



TIBCO iProcess® Objects Director

Administrator's Guide

Version 11.9.1 | March 2024

Contents

Contents	2
Using the TIBCO iProcess Objects Director	3
Overview	3
Multiple Instances of TIBCO iProcess Objects Director	3
Starting and Stopping TIBCO iProcess Objects Director	5
TIBCO iProcess Objects Director Version	5
TIBCO iProcess Objects Server TCP Port Configuration	6
Configuring TIBCO iProcess Objects Director	7
TCP and UDP Ports When Running Multiple Instances of the Director	28
Accessing a TIBCO iProcess Objects Director	29
Connecting to a TIBCO iProcess Objects Server via a Director	31
Director-Related Properties and Methods	33
Pageable Lists/SWXLists of Work Items	37
TIBCO iProcess Objects Director Logging	37
TIBCO Documentation and Support Services	41
Legal and Third-Party Notices	43

Using the TIBCO iProcess Objects Director

This section provides information about using TIBCO iProcess® Objects Director.

Overview

TIBCO iProcess Objects Director is a standalone program that maintains a list of TIBCO iProcess Objects Servers that are configured in a node cluster. When a client needs access to a TIBCO iProcess Objects Server, it first establishes a connection to TIBCO iProcess Objects Director. TIBCO iProcess Objects Director then decides, based on a “pick method,” which TIBCO iProcess Objects Server the client should connect to.

The list of known TIBCO iProcess Objects Servers is updated dynamically as TIBCO iProcess Objects Server instances are started and stopped. The TIBCO iProcess Objects Director maintains this list by checking the `process_config` table of iProcess Engine to which it is associated.

**Note**

For efficiency, after having TIBCO iProcess Objects Director choose a TIBCO iProcess Objects Server for you, you should log in directly to that same TIBCO iProcess Objects Server for all subsequent logins. This is because a SAL session is started for a user the first time they log in; subsequent logins can use that same SAL session, resulting in a much more efficient and faster login.

Multiple Instances of TIBCO iProcess Objects Director

Multiple instances of TIBCO iProcess Objects Director can be run in a node cluster. You may want to run multiple instances of TIBCO iProcess Objects Director on a single machine, each using a different pick method. Or, you may want to run multiple instances on multiple machines for redundancy purposes. Each instance of the Director runs independently from other instances.

Theoretically, you can run up to 99 instances of the Director (the same as for TIBCO iProcess Objects Server), although in reality, it does not seem reasonable nor practical to run that many instances.

Adding and Deleting Instances of TIBCO iProcess Objects Director

When TIBCO iProcess Objects Director is initially installed on a machine, it becomes instance 1 by default. A new installation of a TIBCO iProcess Objects Director on a machine causes an entry to be automatically added to the `process_config` table, as follows:

Machine ID	Process Name	Process Instance
1	DIRECTOR	1

Once the initial installation is completed, additional instances of TIBCO iProcess Objects Director can be added to or deleted from the `process_config` table using the following `swadm` commands:

```
SWDIR\util\swadm add_process MachineID DIRECTOR Y
SWDIR\util\swadm delete_process MachineID DIRECTOR ProcessInst
```

For example, after adding a second instance to machine 1, the `process_config` table appears as follows:

Machine ID	Process Name	Process Instance
1	DIRECTOR	1
1	DIRECTOR	2

For information about using the `swadm` utility, see *TIBCO iProcess Engine Administrator's Guide*.

Process attributes are used to configure TIBCO iProcess Objects Director. Each of the process attributes can be configured differently for each instance of TIBCO iProcess Objects Director when you are running multiple instances of the Director. For instance, each

instance might use a different “pick method” to choose a TIBCO iProcess Objects Server. For information about these process attributes, see [Configuring TIBCO iProcess Objects Director](#).

Starting and Stopping TIBCO iProcess Objects Director

Starting and stopping TIBCO iProcess Objects Director is handled by the Process Manager. Once installed on a node, the TIBCO iProcess Objects Director process is listed in the process configuration table with the process name DIRECTOR. The DIRECTOR process can be started and stopped with all the other SPS processes by using the following commands:

```
SWDIR\bin\swstart
```

```
SWDIR\bin\swstop
```

Or, you can start or stop individual TIBCO iProcess Objects Director instances using the following commands:

```
SWDIR\util\swsvrmgr START MachineID DIRECTOR ProcessInst  
SWDIR\util\swsvrmgr SHUTDOWN MachineID DIRECTOR ProcessInst
```

For information about using the swsvrmgr utility, see *TIBCO iProcess Engine Administrator's Guide*.



Note

You can start or stop individual TIBCO iProcess Objects Director instances using iProcess Administration Console. For more information, see the iProcess Administration Console.

TIBCO iProcess Objects Director Version

You can determine the version of TIBCO iProcess Objects Director by executing the following from the command line:

- `SWDIR\bin\spodirector -v` (Windows) or `$SWDIR/bin/spodirector -v` (UNIX)
- `what $SWDIR/bin/spodirector` (UNIX only)

TIBCO iProcess Objects Server TCP Port Configuration

All TIBCO iProcess Objects Servers that want to make use of TIBCO iProcess Objects Director must be configured to use static TCP ports. This allows TIBCO iProcess Objects Director to be configured with those port numbers so it knows the TCP port number to use when establishing a connection between a client and a TIBCO iProcess Objects Server. For information about configuring TCP ports on TIBCO iProcess Objects Servers, see *TIBCO iProcess Objects Server Administrator's Guide*.

TIBCO iProcess Objects Director determines which TCP port to use by using the `BASE_TCP_SERVICE_NAME` process attribute (see [BASE_TCP_SERVICE_NAME](#)) and the server's instance number.

TIBCO iProcess Objects Director assumes the first instance of each TIBCO iProcess Objects Server uses the TCP port established by `BASE_TCP_SERVICE_NAME` (which defaults to 28021), the second instance uses the *base* port plus 1, the third instance uses the base port plus 2, and so forth.

For example, if `BASE_TCP_SERVICE_NAME` = "DEFAULT" (port 28021) and five TIBCO iProcess Objects Server instances exist on two nodes (two on Node 1 and three on Node 2), the five instances of TIBCO iProcess Objects Server must be configured with the following TCP ports:

Node	Server Instance	TCP Port
1	1	28021
1	2	28022
2	1	28021
2	2	28022
2	3	28023

Configuring TIBCO iProcess Objects Director

TIBCO iProcess Objects Director is configured with the use of process attributes in the Process Manager, using the following command line:

```
SWDIR\util\swadm set_attribute MachID DIRECTOR Proclnst AttrName AttrValue
```

where:

- *MachID* = Machine ID. If 0 (zero) is specified, the attribute is set on all machines in the cluster.
- *Proclnst* = Instance of the TIBCO iProcess Objects Director process. If 0 (zero) is specified, the attribute is set on all instances of TIBCO iProcess Objects Director on the machine specified by *MachID*. (Note - Using 0 (zero) for *Proclnst* for the TCP_SERVICE_NAME or UDP_SERVICE_NAME process attributes causes a “base” TCP or UDP port to be established. This is used to determine the TCP or UDP port for each instance of TIBCO iProcess Objects Director when using multiple instances of TIBCO iProcess Objects Director. For more information, see [TCP and UDP Ports When Running Multiple Instances of the Director.](#))
- *AttrName* = Name of the process attribute.
- *AttrValue* = Value to assign to the process attribute.



Note

If an invalid value is specified for a TIBCO iProcess Objects Director process attribute, the attribute is set to its default value, plus an entry is written to the Windows Event Log (Windows systems) or the \$SWDIR/logs/dir_error file (UNIX systems).

Also, if you delete an instance-specific TIBCO iProcess Objects Director process attribute (with delete_attribute), the process attribute value reverts to the default value for that attribute (see the table on the following pages for the default values for each attribute). Note that this only applies to the process attributes that can be changed while TIBCO iProcess Objects Director is running.

TIBCO iProcess Objects Director Process Attributes

The following table lists the process attributes that are used to control TIBCO iProcess Objects Director.

Process Attribute	Description
PICK_METHOD Type: Integer Range: 1 to 7 Default: 1	<p>This is the method TIBCO iProcess Objects Director uses when selecting a TIBCO iProcess Objects Server to connect the client to. They are:</p> <p>1 - Random</p> <p>TIBCO iProcess Objects Director randomly selects one of TIBCO iProcess Objects Servers.</p> <p>2 - Round-Robin</p> <p>TIBCO iProcess Objects Director selects TIBCO iProcess Objects Servers in the order in which they appear in TIBCO iProcess Objects Director's list of known Servers.</p> <p>3 - Loaded Random</p> <p>Each TIBCO iProcess Objects Server is given a certain percentage of the connections, based on a random number between 1 and 10 and the load factor that you've specified using the LOAD_BALANCE process attribute.</p> <p>For example, suppose two TIBCO iProcess Objects Servers exist with specified load balances of 30% and 70%. If the random number falls between 1 and 3, the first TIBCO iProcess Objects Server (the one with a load balance of 30%) is given the connection; if the random number falls between 4 and 10, the second TIBCO iProcess Objects Server is given the connection.</p> <p>The LOAD_BALANCE attribute must be specified if using this pick method.</p>
PICK_METHOD (Cont.)	<p>4 - Connection Count Over a Period</p> <p>Each TIBCO iProcess Objects Server is given connections, based on the following factors:</p>

Process Attribute	Description
	<ul style="list-style-type: none"> the load factor that you've specified using the LOAD_BALANCE process attribute, and the number of connections each TIBCO iProcess Objects Server has received during the period specified by the DELTA_LENGTH process attribute (which defaults to the previous 600 seconds). <p>TIBCO iProcess Objects Director determines the number of connections each TIBCO iProcess Objects Server has received during the delta period. It converts that number of connections to a percentage of the total number of connections for all TIBCO iProcess Objects Servers during the delta period. It then attempts to match that percentage with the percentage specified in the LOAD_BALANCE attribute, giving connections to TIBCO iProcess Objects Server that is furthest away from the percentage you've specified for that TIBCO iProcess Objects Server in the LOAD_BALANCE attribute.</p> <p>The connection statistics are updated periodically in the number of seconds specified in the SERVER_CHECK_PERIOD process attribute. The variance from the percentage specified by LOAD_BALANCE depends on how often the statistical information is checked.</p> <p>The LOAD_BALANCE attribute must be specified if using this pick method.</p>

PICK_METHOD (Cont.)

5 - Total Connection Count

Each TIBCO iProcess Objects Server is given connections, based on the following factors:

- the load factor that you've specified using the LOAD_BALANCE process attribute, and
- the total number of connections each TIBCO iProcess Objects Server has received since the server was started.

TIBCO iProcess Objects Director determines the total number of

Process Attribute	Description
	<p>connections each TIBCO iProcess Objects Server has received since it was started. It converts that number of connections to a percentage of the total number of connections for all TIBCO iProcess Objects Servers.</p> <p>It then attempts to match that percentage with the percentage specified in the LOAD_BALANCE attribute, giving connections to TIBCO iProcess Objects Server that is furthest away from the percentage you've specified for that TIBCO iProcess Objects Server in the LOAD_BALANCE attribute.</p> <p>The connection statistics are updated periodically in the number of seconds specified in the SERVER_CHECK_PERIOD process attribute. The variance from the percentage specified by LOAD_BALANCE depends on how often the statistical information is checked.</p> <p>The LOAD_BALANCE attribute must be specified if using this pick method.</p>

PICK_METHOD (Cont.)

6 - Transaction Count Over a Period

Each TIBCO iProcess Objects Server is given connections, based on the following factors:

- the load factor that you've specified using the LOAD_BALANCE process attribute, and
- the number of transactions each TIBCO iProcess Objects Server has processed during the period specified by the DELTA_LENGTH process attribute (which defaults to the previous 600 seconds).

TIBCO iProcess Objects Director determines the number of transactions each TIBCO iProcess Objects Server has processed during the delta period. It converts that number of transactions to a percentage of the total number of transactions for all TIBCO iProcess Objects Servers during the delta period. It then attempts to match that percentage with the percentage specified in the LOAD_BALANCE attribute, giving connections to TIBCO iProcess

Process Attribute	Description
	<p>Objects Server that is furthest away from the percentage you've specified for that TIBCO iProcess Objects Server in the LOAD_BALANCE attribute.</p> <p>The transaction statistics are updated periodically in the number of seconds specified in the SERVER_CHECK_PERIOD process attribute. The variance from the percentage specified by LOAD_BALANCE depends on how often the statistical information is checked.</p> <p>Note: A transaction consists of a request from the client to TIBCO iProcess Objects Server, and the response from TIBCO iProcess Objects Server back to the client. It is synonymous with "message" request/response (it has nothing to do with database transactions). This pick method allows you to balance load based on network traffic (for example, messages) rather than connections.</p> <p>The LOAD_BALANCE attribute must be specified if using this pick method.</p>

PICK_METHOD (Cont.)

7 - Total Transaction Count

Each TIBCO iProcess Objects Server is given connections, based on the following factors:

- the load factor that you've specified using the LOAD_BALANCE process attribute, and
- the total number of transactions each TIBCO iProcess Objects Server has processed since the server was started.

TIBCO iProcess Objects Director determines the total number of transactions each TIBCO iProcess Objects Server has processed since the server was started. It converts that number of transactions to a percentage of the total number of transactions for all TIBCO iProcess Objects Servers. It then attempts to match that percentage with the percentage specified in the LOAD_BALANCE attribute, giving connections to TIBCO iProcess Objects Server that is furthest away from the percentage you've specified

Process Attribute	Description
	<p>for that TIBCO iProcess Objects Server in the LOAD_BALANCE attribute.</p> <p>The transaction statistics are updated periodically in the number of seconds specified in the SERVER_CHECK_PERIOD process attribute. The variance from the percentage specified by LOAD_BALANCE depends on how often the statistical information is checked.</p> <p>Note: A transaction consists of a request from the client to TIBCO iProcess Objects Server, and the response from TIBCO iProcess Objects Server back to the client. It is synonymous with "message" request/response (it has nothing to do with database transactions). This pick method allows you to balance load based on network traffic (i.e., messages) rather than connections. The total transaction count for a TIBCO iProcess Objects Server can be determined using the TotTransCnt property on the SWNodeInfoEx object (TIBCO iProcess Objects), or the getTotalTransCnt method on the vANode object (TIBCO iProcess Server Objects).</p> <p>The LOAD_BALANCE attribute must be specified if using this pick method.</p> <p>This process attribute can be changed while TIBCO iProcess Objects Director is running.</p>
LOAD_BALANCE Type: String Range: N/A Default: Undefined	<p>This attribute allocates a load to each TIBCO iProcess Objects Server in the cluster. It calculates a percentage for each instance based on the total you've specified.</p> <p>Format : <i>MachineID InstanceNum=Load</i></p> <p>For example:</p> <pre>1 1=40,1 2=40,2 1=120</pre> <p>allocates the load as follows:</p> <ul style="list-style-type: none"> Node 1, TIBCO iProcess Objects Server Instance 1 = 20% of load

Process Attribute	Description
	<ul style="list-style-type: none"> Node 1, TIBCO iProcess Objects Server Instance 2 = 20% of load Node 2, TIBCO iProcess Objects Server Instance 1 = 60% of load <p>TIBCO iProcess Objects Director recalculates the load percentages, if TIBCO iProcess Objects Server becomes unavailable. So, if in our example, server Instance 1 on Node 2 became unavailable, the percentages would be recalculated as follows:</p> <ul style="list-style-type: none"> Node 1, TIBCO iProcess Objects Server Instance 1 = 50% of load Node 1, TIBCO iProcess Objects Server Instance 2 = 50% of load Node 2, TIBCO iProcess Objects Server Instance 1 = 0% of load <p>If a load balance is not specified for a known TIBCO iProcess Objects Server, it defaults to 0 (zero); it will not be given connections.</p> <p>This process attribute must be specified when using pick methods 3-7.</p> <p>This process attribute can be changed while TIBCO iProcess Objects Director is running.</p>
LOG_LEVEL Type: Integer Range: 1 to 4 Default: 2	<p>Level of information written to the TIBCO iProcess Objects Director log file. The possible values are:</p> <ul style="list-style-type: none"> 1 - Errors 2 - Errors and Warnings 3 - Errors, Warnings, and Information 4 - Errors, Warnings, Information, and Debug <p>Note that if this attribute is set to level 4 (Debug), request/response messages are automatically written to the log, regardless of the setting of the TRACE_MSG attribute.</p>

Process Attribute	Description
	This process attribute can be changed while TIBCO iProcess Objects Director is running.
LOG_FILE_MAX_SIZE Type: Integer Range: 1 to 9999 MB Default: 15	The maximum size in MB of the TIBCO iProcess Objects Director log file before it is rolled over. This process attribute can be changed while TIBCO iProcess Objects Director is running.
LOG_FILE_MAX_ARCHIVES Type: Integer Range: 0 to 99999 Default: 0	The maximum number of archived log files created if the log rolls over. A value of 0 means, do not archive logs. This process attribute can be changed while TIBCO iProcess Objects Director is running.
TRACE_MSG Type: Integer Range: 0 (no) or 1 (yes) Default: 0	A flag that specifies if the client request and the response messages should be written to the log file. Note that request/response messages are written to the log if the LOG_LEVEL attribute is set to 4 (Debug), regardless of the setting of this attribute. This process attribute can be changed while TIBCO iProcess Objects Director is running.
LOG_CATEGORIES Type: String Range: N/A Default: 0xFFFFFFFF	A set of bit flags to indicate which logging areas should be switched on. An individual category may be specified, or you can combine the category values, then set this attribute to the calculated value. LOGCAT_ALL 0xFFFFFFFF LOGCAT_MAIN_THREAD 0x00000001 LOGCAT_STATUS_UPDATE_THREAD 0x00000002 LOGCAT_PERIODIC_STATS_THREAD 0x00000004 LOGCAT_STATS_SEND_THREAD 0x00000008

Process Attribute	Description
	LOGCAT_STATS_RECV_THREAD 0x00000010
	LOGCAT_RECV_UDP_THREAD 0x00000020
	LOGCAT_MESSAGE_SEND_THREAD 0x00000040
	LOGCAT_MESSAGE_RECV_THREAD 0x00000080
	LOGCAT_GET_ONE_NODE 0x00000100
	LOGCAT_GET_ALL_NODES 0x00000200
	LOGCAT_LOAD_BALANCE_THREAD 0x00000400
	LOGCAT_LOG 0x01000000
	LOGCAT_SAL_TIMING 0x02000000
	This process attribute can be changed while TIBCO iProcess Objects Director is running.
NUM_THREADS Type: Integer Range: 1 to 512 Default: 5	The number of threads TIBCO iProcess Objects Director has for processing client requests. This process attribute cannot be changed while TIBCO iProcess Objects Director is running.
MESSAGE_TIMEOUT Type: Integer Range: 1 to 300 secs. Default: 15	The time in seconds that TIBCO iProcess Objects Director waits for a socket response from one of the TIBCO iProcess Objects Servers after requesting a statistics update. If a response is not received, the request is resent up to a maximum number of times indicated by MAX_SOCKET_ATTEMPTS. This process attribute can be changed while TIBCO iProcess Objects Director is running.
TCP_MAX_CLIENTS Type: Integer Range: 0 to 999999 Default: 1024	Used only on UNIX systems. This is used to allocate the proper amount of memory. Do not change the value of this attribute unless instructed by TIBCO Technical Support. This process attribute cannot be changed while TIBCO iProcess

Process Attribute	Description
	Objects Director is running.
MAX_SOCKET_ATTEMPTS Type: Integer Range: 1 to 50 Default: 3	<p>The maximum number of attempts to send a statistics message to a TIBCO iProcess Objects Server. Also, the maximum number of attempts to receive a response from a successfully sent statistics message.</p> <p>This is also the number of times TIBCO iProcess Objects Director attempts to connect to an iProcess Objects Server. It tries the specified number of times, wait for the period specified by the SERVER_CHECK_PERIOD attribute, then try this number of times again, indefinitely (or until the Process Manager notifies the TIBCO iProcess Objects Director that the iProcess Objects Server is no longer running).</p> <p>This process attribute can be changed while the TIBCO iProcess Objects Director is running.</p>
STICKY_SAL Type: Integer Range: 0 (no) or 1 (yes) Default: 1	<p>This flag specifies if TIBCO iProcess Objects Director should allocate the user to the TIBCO iProcess Objects Server instance that has already opened a SAL session for this user.</p> <p>If the value of this parameter is set to 0, TIBCO iProcess Objects Director chooses a new TIBCO iProcess Objects Server instance that has a new SAL session started. If the value of this parameter is set to 1, the TIBCO iProcess Objects Server instance that has opened a SAL session for this user is allocated to the user.</p> <p>Note: This attribute is not defined on a newly installed TIBCO iProcess Objects Director. To use this attribute, you must explicitly assign a value to it using the <code>swadm set_attribute</code> command. For more information about this command, see "Set a Process Attribute" in <i>TIBCO iProcess Engine Administrator's Guide</i>.</p>
BASE_TCP_SERVICE_NAME Type: String Range: NA	<p>Specifies the "base" TCP port for TIBCO iProcess Objects Servers that uses TIBCO iProcess Objects Director. TIBCO iProcess Objects Director uses this to determine the port on which to establish the connection between a client and TIBCO iProcess Objects Server. It's also used to send statistics messages to TIBCO iProcess</p>

Process Attribute	Description
Default: DEFAULT	<p>Objects Server instances.</p> <p>The first instance of each TIBCO iProcess Objects Server uses the “base” TCP port, the second instance uses the “base” TCP port plus 1, and so on.</p> <p>The base TCP port can be specified in the following ways:</p> <ul style="list-style-type: none"> Specify a value of “DEFAULT” (the default). This causes the base TCP port to be established as 28021. Specify the actual base TCP port number. Specify a “service name” that can map to the TCP port number. This requires that you add the service name to the %SystemRoot%\System32\Drivers\Etc\Services file (Windows) or /etc/services file (UNIX) that maps the service name to the TCP port used as the base TCP port. <p>For more information, see TIBCO iProcess Objects Server TCP Port Configuration.</p> <p>This process attribute cannot be changed while the TIBCO iProcess Objects Director is running.</p>
TCP_SERVICE_NAME Type: String Range: NA Default: DEFAULT	<p>Identifies the port number on which TIBCO iProcess Objects Director listens for client connections. This can be specified in the following ways:</p> <ul style="list-style-type: none"> Specify a value of “DEFAULT”. This means, use a dynamic port, which causes the operating system to assign the port number when the TIBCO iProcess Objects Director starts. This designation is used if you are issuing a UDP broadcast to determine the available TIBCO iProcess Objects Directors, or you are issuing a directed UDP message to a specific TIBCO iProcess Objects Director. Specify a value other than “DEFAULT”. This means, use a static port, which causes the TCP port number to be fixed for the TIBCO iProcess Objects Director you are configuring. This is used if you are manually creating the node object

Process Attribute	Description
	<p>that represents the TIBCO iProcess Objects Director (which requires that you know the TCP port the TIBCO iProcess Objects Director is using).</p> <p>To configure the TIBCO iProcess Objects Director to use a static TCP port, either specify the desired TCP port number in this process attribute or specify a “service name” that can map to the TCP port number. If using a service name, you must add the service name to the %SystemRoot%\System32\Drivers\Etc\Services file (Windows) or /etc/services file (UNIX) that maps the service name to the TCP port on which you want the TIBCO iProcess Objects Director to listen for client connections.</p> <p>For information about configuring TCP ports when running multiple instances of the TIBCO iProcess Objects Director, see TCP and UDP Ports When Running Multiple Instances of the Director.</p> <p>This process attribute cannot be changed while the TIBCO iProcess Objects Director is running.</p>
TCP_RESPONSE_PAGES Type: Integer Range: 1 to 8 pages Default: 1	<p>Size of the TCP response buffer in pages. Each page is 2k bytes.</p> <p>This process attribute cannot be changed while the TIBCO iProcess Objects Director is running.</p>
TCP_Q_LENGTH Type: Integer Range: 0 to 999999 Default: 0	<p>The number of TCP connection requests to allow the TCP kernel to queue up. The default of 0 (zero) means the maximum allowed by the kernel.</p> <p>This process attribute cannot be changed while the TIBCO iProcess Objects Director is running.</p>
TCP_RESOLVE_NAME Type: Integer Range: 0 (no) or 1 (yes)	<p>This flag specifies if the TIBCO iProcess Objects Director should employ TCP name resolution (DNS, host file, YP, etc.) for all client connection requests. If set to 0, the TIBCO iProcess Objects Director uses the client IP address and TCP connection port to identify clients.</p>

Process Attribute	Description
<p>Default: 0</p>	<p>This parameter is used for debugging purposes. When set to 1, the machine name of the client appears in the log file, instead of the client's IP address.</p> <p>This process attribute can be changed while the TIBCO iProcess Objects Director is running.</p>
<p>UDP_SERVICE_NAME</p> <p>Type: String</p> <p>Range: NA</p> <p>Default: DEFAULT</p>	<p>Identifies the port number on which TIBCO iProcess Objects Director listens for UDP messages/broadcasts. This can be specified in one of the following ways:</p> <ul style="list-style-type: none"> Specify a value of "DEFAULT" (the default). This causes the TIBCO iProcess Objects Director to listen for UDP broadcasts/messages on port 28001. Specify the port number on which you want the TIBCO iProcess Objects Director to listen for UDP messages/broadcasts. Specify a "service name" that can map to the UDP port number. This requires that you add the service name to the %SystemRoot%\System32\Drivers\Etc\Services file (Windows) or /etc/services file (UNIX) that maps the service name to the UDP port on which you want the TIBCO iProcess Objects Director to listen for UDP messages/broadcasts. Specify "None". This causes TIBCO iProcess Objects Director to not open a UDP port. For example, it will not respond to UDP messages/broadcasts. <p>For information about configuring UDP ports when running multiple instances of the TIBCO iProcess Objects Director, see TCP and UDP Ports When Running Multiple Instances of the Director.</p> <p>This process attribute cannot be changed while the TIBCO iProcess Objects Director is running.</p>
<p>UDP_SERVICE_DESC</p> <p>Type: String</p>	<p>UDP service description that TIBCO iProcess Objects Director sends in response to UDP broadcasts. This can be any string that can be</p>

Process Attribute	Description
Range: NA Default: TIBCO iProcess Objects Director	<p>useful to the client to identify the TIBCO iProcess Objects Director. This description is available in the SWNodeInfo.SWEOSrvDesc property (TIBCO iProcess Objects), or with the getSEOSrvDesc method on vNode (TIBCO iProcess Server Objects).</p> <p>This process attribute cannot be changed while the TIBCO iProcess Objects Director is running.</p>
SERVER_CHECK_PERIOD Type: Integer Range: 1 - 3600 secs. Default: 30	<p>The period in seconds that the TIBCO iProcess Objects Director waits between checking statistical information (number of connections, number of transactions) on the iProcess Objects Servers that are running. The TIBCO iProcess Objects Director uses this statistical information for pick methods 4-7.</p> <p>This is also the period that TIBCO iProcess Objects Director waits between attempts to connect to an iProcess Objects Server. It tries to connect the number of times specified by the MAX_SOCKET_ATTEMPTS attribute, wait for the specified time, then attempt again, indefinitely (or until the Process Manager notifies the TIBCO iProcess Objects Director that the iProcess Objects Server is no longer running).</p> <p>This process attribute can be changed while the TIBCO iProcess Objects Director is running.</p>
DELTA_LENGTH Type: Integer Range: 60 – 3600 secs. Default: 600	<p>The amount of time taken in the past to check for the number of connections and transactions that have occurred during that period. This time is used when the PICK_METHOD attribute is set to 4 or 6 (“connection count over a period” and “transaction count over a period”). For example, if the delta length is 600 seconds (the default), these pick methods look at the number of connections or transactions over the last 600 seconds to determine which TIBCO iProcess Objects Server should receive the next connection.</p> <p>This process attribute can be changed while the TIBCO iProcess Objects Director is running.</p>
WRITEERRSTOEVENTLOG	Specifies whether or not to write errors to the system log (in UNIX

Process Attribute	Description
Type: Integer Range: 0 (no) or 1 (yes) Default: 0	systems) and the event log (in Windows systems). This process attribute can be changed while TIBCO iProcess Objects Director is running.
IDENTIFY_SPO_MACHINE_BY Type: String Range: N/A Default: HOSTNAME	Specifies which TIBCO iProcess Objects Server the client connects to in two ways. They are: <ul style="list-style-type: none"> • HOSTNAME: the machine name of the TIBCO iProcess Objects Server node you want to connect to. • IP: the IP address of the TIBCO iProcess Objects Server node you want to connect to.
LOG_CATEGORIES Type: String Range: N/A Default: 0xFFFFFFFF	A set of bit flags to indicate which logging areas should be switched on. An individual category may be specified, or you can combine the category values, then set this attribute to the calculated value. LOGCAT_ALL 0xFFFFFFFF LOGCAT_MAIN_THREAD 0x00000001 LOGCAT_STATUS_UPDATE_THREAD 0x00000002 LOGCAT_PERIODIC_STATS_THREAD 0x00000004 LOGCAT_STATS_SEND_THREAD 0x00000008 LOGCAT_STATS_RECV_THREAD 0x00000010 LOGCAT_RECV_UDP_THREAD 0x00000020 LOGCAT_MESSAGE_SEND_THREAD 0x00000040 LOGCAT_MESSAGE_RECV_THREAD 0x00000080 LOGCAT_GET_ONE_NODE 0x00000100 LOGCAT_GET_ALL_NODES 0x00000200 LOGCAT_LOAD_BALANCE_THREAD 0x00000400 LOGCAT_LOG 0x01000000

Process Attribute	Description
	LOGCAT_SAL_TIMING 0x02000000 This process attribute can be changed while the TIBCO iProcess Objects Director is running.
NUM_THREADS Type: Integer Range: 1 to 512 Default: 5	The number of threads the TIBCO iProcess Objects Director has for processing client requests. This process attribute cannot be changed while the TIBCO iProcess Objects Director is running.
MESSAGE_TIMEOUT Type: Integer Range: 1 to 300 secs. Default: 15	The time in seconds the TIBCO iProcess Objects Director will wait for a socket response from one of the TIBCO iProcess Objects Servers after requesting a statistics update. If a response is not received, the request is resent up to a maximum number of times indicated by MAX_SOCKET_ATTEMPTS (see). This process attribute can be changed while the TIBCO iProcess Objects Director is running.
TCP_MAX_CLIENTS Type: Integer Range: 0 to 999999 Default: 1024	Used only on UNIX systems. This is used to allocate the proper amount of memory. Do not change the value of this attribute unless instructed by TIBCO Technical Support. This process attribute cannot be changed while the TIBCO iProcess Objects Director is running.
MAX_SOCKET_ATTEMPTS Type: Integer Range: 1 to 50 Default: 3	Maximum number of attempts to send a statistics message to a TIBCO iProcess Objects Server. Also, the maximum number of attempts to receive a response from a successfully sent statistics message. This is also the number of times the TIBCO iProcess Objects Director will attempt to connect to an iProcess Objects Server. It will try this number of times, wait the period of time specified by the SERVER_CHECK_PERIOD attribute, then try this number of times again, indefinitely (or until the Process Manager notifies the TIBCO iProcess Objects Director that the iProcess Objects Server is

Process Attribute	Description
	no longer running).
	This process attribute can be changed while the TIBCO iProcess Objects Director is running.
STICKY_SAL Type: Integer Range: 0 (no) or 1 (yes) Default: 1	<p>This flag specifies if TIBCO iProcess Objects Director should allocate the user to the TIBCO iProcess Objects Server instance that has already opened a SAL session for this user.</p> <p>If the value of this parameter is set to 0, TIBCO iProcess Objects Director chooses a new TIBCO iProcess Objects Server instance that has a new SAL session started. If the value of this parameter is set to 1, the TIBCO iProcess Objects Server instance that has opened a SAL session for this user is allocated to the user.</p> <p>Note: This attribute is not defined on a newly installed TIBCO iProcess Objects Director. To use this attribute, you must explicitly assign a value to it using the <code>swadm set_attribute</code> command. For more information about this command, see "Set a Process Attribute" in <i>TIBCO iProcess Engine Administrator's Guide</i>.</p>
BASE_TCP_SERVICE_NAME Type: String Range: NA Default: DEFAULT	<p>Specifies the "base" TCP port for TIBCO iProcess Objects Servers that will be using this TIBCO iProcess Objects Director. The TIBCO iProcess Objects Director uses this to determine the port on which to establish the connection between a client and TIBCO iProcess Objects Server. It's also used to send statistics messages to TIBCO iProcess Objects Server instances.</p> <p>The first instance of each TIBCO iProcess Objects Server uses the "base" TCP port, the second instance uses the "base" TCP port plus 1, and so on.</p> <p>The base TCP port can be specified in the following ways:</p> <ul style="list-style-type: none"> Specify a value of "DEFAULT" (the default). This causes the base TCP port to be established as 28021. Specify the actual base TCP port number. Specify a "service name" that will map to the TCP port number. This requires that you add the service name to the

Process Attribute	Description
	<p>%SystemRoot%\System32\Drivers\Etc\Services file (Windows) or /etc/services file (UNIX) that maps the service name to the TCP port used as the base TCP port.</p> <p>For more information, see TIBCO iProcess Objects Server TCP Port Configuration.</p> <p>This process attribute cannot be changed while the TIBCO iProcess Objects Director is running.</p>
TCP_SERVICE_NAME Type: String Range: NA Default: DEFAULT	<p>Identifies the port number on which the TIBCO iProcess Objects Director will listen for client connections. This can be specified in the following ways:</p> <ul style="list-style-type: none"> Specify a value of “DEFAULT”. This means use a dynamic port, which causes the operating system to assign the port number when the TIBCO iProcess Objects Director starts. This designation is used if you are issuing a UDP broadcast to determine the available TIBCO iProcess Objects Directors, or you are issuing a directed UDP message to a specific TIBCO iProcess Objects Director. Specify a value other than “DEFAULT”. This means use a static port, which causes the TCP port number to be fixed for the TIBCO iProcess Objects Director you are configuring. This is used if you are manually creating the node object that represents the TIBCO iProcess Objects Director (which requires that you know the TCP port the TIBCO iProcess Objects Director is using). <p>To configure the TIBCO iProcess Objects Director to use a static TCP port, either specify the desired TCP port number in this process attribute, or specify a “service name” that will map to the TCP port number. If using a service name, you must add the service name to the %SystemRoot%\System32\Drivers\Etc\Services file (Windows) or /etc/services file (UNIX) that maps the service name to the TCP port on which you want the TIBCO iProcess Objects Director to listen for client connections.</p>

Process Attribute	Description
	<p>For information about configuring TCP ports when running multiple instances of the TIBCO iProcess Objects Director, see TCP and UDP Ports When Running Multiple Instances of the Director.</p> <p>This process attribute cannot be changed while the TIBCO iProcess Objects Director is running.</p>

Process Attribute	Description
TCP_RESPONSE_PAGES Type: Integer Range: 1 to 8 pages Default: 1	<p>Size of the TCP response buffer in pages. Each page is 2k bytes.</p> <p>This process attribute cannot be changed while TIBCO iProcess Objects Director is running.</p>
TCP_Q_LENGTH Type: Integer Range: 0 to 999999 Default: 0	<p>Number of TCP connection requests to allow the TCP kernel to queue up. The default of 0 (zero) means the maximum allowed by the kernel.</p> <p>This process attribute cannot be changed while TIBCO iProcess Objects Director is running.</p>
TCP_RESOLVE_NAME Type: Integer Range: 0 (no) or 1 (yes) Default: 0	<p>This flag specifies if TIBCO iProcess Objects Director should employ TCP name resolution (DNS, host file, YP, etc.) for all client connection requests. If set to 0, TIBCO iProcess Objects Director uses the client IP address and TCP connection port to identify clients.</p> <p>This parameter is used for debugging purposes. When set to 1, the machine name of the client appears in the log file, instead of the client's IP address.</p> <p>This process attribute can be changed while TIBCO iProcess Objects Director is running.</p>
UDP_SERVICE_NAME	Identifies the port number on which TIBCO iProcess Objects

Process Attribute	Description
Type: String Range: NA Default: DEFAULT	<p>Director will listen for UDP messages/broadcasts. This can be specified in one of the following ways:</p> <ul style="list-style-type: none"> Specify a value of “DEFAULT” (the default). This causes TIBCO iProcess Objects Director to listen for UDP broadcasts/messages on port 28001. Specify the port number on which you want TIBCO iProcess Objects Director to listen for UDP messages/broadcasts. Specify a “service name” that will map to the UDP port number. This requires that you add the service name to the %SystemRoot%\System32\Drivers\Etc\Services file (Windows) or /etc/services file (UNIX) that maps the service name to the UDP port on which you want TIBCO iProcess Objects Director to listen for UDP messages/broadcasts. Specify “None”. This causes TIBCO iProcess Objects Director to not open a UDP port, for example, it will not respond to UDP messages/broadcasts. <p>For information about configuring UDP ports when running multiple instances of TIBCO iProcess Objects Director, see TCP and UDP Ports When Running Multiple Instances of the Director.</p> <p>This process attribute cannot be changed while TIBCO iProcess Objects Director is running.</p>
UDP_SERVICE_DESC Type: String Range: NA Default: TIBCO iProcess Objects Director	<p>UDP service description that TIBCO iProcess Objects Director will send in response to UDP broadcasts. This can be any string that will be useful to the client to identify TIBCO iProcess Objects Director. This description is available in the SWNodeInfo.SWEOSrvDesc property (TIBCO iProcess Objects), or with the getSEOSrvDesc method on vNode (TIBCO iProcess Server Objects).</p> <p>This process attribute cannot be changed while TIBCO iProcess Objects Director is running.</p>

Process Attribute	Description
SERVER_CHECK_PERIOD Type: Integer Range: 1 - 3600 secs. Default: 30	<p>The time in seconds that TIBCO iProcess Objects Director waits between checking statistical information (number of connections, number of transactions) on the iProcess Objects Servers that are running. TIBCO iProcess Objects Director uses this statistical information for pick methods 4-7.</p> <p>This is also the amount of time that TIBCO iProcess Objects Director will wait between attempts to connect to an iProcess Objects Server. It will try to connect the number of times specified by the MAX_SOCKET_ATTEMPTS attribute, wait for the specified time, then attempt again, indefinitely (or until the Process Manager notifies TIBCO iProcess Objects Director that iProcess Objects Server is no longer running).</p> <p>This process attribute can be changed while TIBCO iProcess Objects Director is running.</p>
DELTA_LENGTH Type: Integer Range: 60 – 3600 secs. Default: 600	<p>The time taken in the past to check the number of connections and transactions that have occurred during that period. This duration is used when the PICK_METHOD attribute is set to 4 or 6 (“connection count over a period” and “transaction count over a period”). For example, if the delta length is 600 seconds (the default), these pick methods will look at the number of connections or transactions over the last 600 seconds to determine which TIBCO iProcess Objects Server should receive the next connection.</p> <p>This process attribute can be changed while TIBCO iProcess Objects Director is running.</p>
WRITEERRSTOEVENTLOG Type: Integer Range: 0 (no) or 1 (yes) Default: 0	<p>Specifies whether or not to write errors to the system log (in UNIX systems) and the event log (in Windows systems).</p> <p>This process attribute can be changed while TIBCO iProcess Objects Director is running.</p>
IDENTIFY_SPO_MACHINE_BY	<p>Specifies which TIBCO iProcess Objects Server the client connects to in two ways. They are:</p>

Process Attribute	Description
Type: String	<ul style="list-style-type: none"> • HOSTNAME: the machine name of the TIBCO iProcess Objects Server node you want to connect to. • IP: the IP address of the TIBCO iProcess Objects Server node you want to connect to.
Range: N/A	
Default: HOSTNAME	

TCP and UDP Ports When Running Multiple Instances of the Director

When running multiple instances of TIBCO iProcess Objects Director, the TCP_SERVICE_NAME and UDP_SERVICE_NAME process attributes work somewhat differently than the other process attributes, as follows:

TCP_SERVICE_NAME

The following is the process that TIBCO iProcess Objects Director goes through to establish a TCP port when it starts up:

1. TIBCO iProcess Objects Director looks to see if an instance-specific TCP_SERVICE_NAME process attribute is defined for the instance of TIBCO iProcess Objects Director that is starting. For example, if instance 2 of TIBCO iProcess Objects Director is starting, it looks to see if `set_attribute` was executed for TCP_SERVICE_NAME using a *Proclnst* of 2. If an instance-specific TCP_SERVICE_NAME was defined, it uses the TCP port specified.
2. If an instance-specific TCP_SERVICE_NAME process attribute has *not* been defined, TIBCO iProcess Objects Director looks to see if a “base” TCP_SERVICE_NAME process attribute has been defined (`set_attribute` was executed using a *Proclnst* of 0). If a base TCP_SERVICE_NAME process attribute has been defined, it adds the instance number (minus 1; because the base number is used by instance 1) to the base TCP port number to determine the TCP port for that TIBCO iProcess Objects Director (for example, if the base TCP port number is 10000, instance 3 of TIBCO iProcess Objects Director will use TCP port 10002).
3. If neither the instance-specific nor the “base” TCP_SERVICE_NAME process attribute has been defined, it defaults to dynamic, causing the operating system to assign the port number when TIBCO iProcess Objects Director is started.

For more information about specifying the TCP port, see [TCP_SERVICE_NAME](#) process attribute.

UDP_SERVICE_NAME

The following is the process a TIBCO iProcess Objects Director goes through to establish a UDP port when it starts up:

1. The TIBCO iProcess Objects Director looks to see if an instance-specific `UDP_SERVICE_NAME` process attribute is defined for the instance of TIBCO iProcess Objects Director that is starting. For example, if instance 2 of TIBCO iProcess Objects Director is starting, it looks to see if `set_attribute` was executed for `UDP_SERVICE_NAME` using a *Proclnst* of 2. If the instance-specific `UDP_SERVICE_NAME` was defined, it uses the UDP port specified (or if “None” is specified, no UDP port opens).
2. If an instance-specific `UDP_SERVICE_NAME` process attribute has not been defined, TIBCO iProcess Objects Director looks to see if a “base” `UDP_SERVICE_NAME` process attribute has been defined (`set_attribute` was executed using a *Proclnst* of 0). If a base `UDP_SERVICE_NAME` process attribute has been defined, it adds the instance number (minus 1; because the base number is used by instance 1) to the base UDP port number to determine the UDP port for that TIBCO iProcess Objects Director (for example, if the base UDP port is 55670, instance 3 of TIBCO iProcess Objects Director uses port number 55672).
3. If neither the instance-specific nor the “base” `UDP_SERVICE_NAME` process attribute has been defined, the first instance of TIBCO iProcess Objects Director is assigned port 28001 (the default for TIBCO iProcess Objects Directors), instance 2 is assigned 28002, and so on.

For more information about specifying the UDP port, see [UDP_SERVICE_NAME](#) process attribute.

Accessing a TIBCO iProcess Objects Director

How you access a TIBCO iProcess Objects Director depends on whether you are using TIBCO iProcess Objects or TIBCO iProcess Server Objects, as follows:

TIBCO iProcess Objects

A TIBCO iProcess Objects client accesses TIBCO iProcess Objects Directors using one of the following methods:

- **Auto-discovery UDP Broadcast** The client can send out a UDP broadcast to auto-discover TIBCO iProcess Objects Directors on the LAN segment. By default, TIBCO iProcess Objects Directors listen for UDP broadcasts on UDP port 28001. You can change this for a TIBCO iProcess Objects Director using the `UDP_SERVICE_NAME` process attribute (see [UDP_SERVICE_NAME](#)). For each TIBCO iProcess Objects Director that responds to the broadcast, an `SWNodeInfo` object is added to the `SWEnterprise.NodeInfos` list.

For more information about UDP broadcasts, see *TIBCO iProcess Objects Programmer's Guide*.

- **Directed UDP Message** Using the `AddNode` or `AddNodeEx` method, the client can direct a UDP message to a TIBCO iProcess Objects Director (`AddNodeEx` allows you to direct the UDP to a specific instance of the Director). If directed to a TIBCO iProcess Objects Director, the `AddNode/AddNodeEx` method's `IsDirector` parameter must be set to `True`. If TIBCO iProcess Objects Director responds to the UDP message, a `SWNodeInfo` object representing TIBCO iProcess Objects Director is added to the `SWEnterprise.NodeInfos` list.

For more information about directed UDP messages, see *TIBCO iProcess Objects Programmer's Guide*.

- **Manually Creating an SWNodeInfo Object** Using the `MakeNodeInfo`, `MakeNodeInfoEx`, or `MakeNodeInfoByTag` method, the client can manually add the `SWNodeInfo` object to the `NodeInfos` list (`MakeNodeInfoEx` allows you to add a specific instance of TIBCO iProcess Objects Director). If the `SWNodeInfo` object is to represent a TIBCO iProcess Objects Director, the `MakeNodeInfo/MakeNodeInfoEx` method's `IsDirector` parameter must be set to `True` when it is called.

For more information about manually creating an `SWNodeInfo` object, see *TIBCO iProcess Objects Programmer's Guide*.

TIBCO iProcess Server Objects

A TIBCO iProcess Server Objects client accesses TIBCO iProcess Objects Directors using one of the following methods:

- **Auto-discovery UDP Broadcast** The client can send out a UDP broadcast to auto-discover TIBCO iProcess Objects Directors on the LAN segment by calling the

getNodes method on sNodeManager. This method returns an array of vNode objects, one for each TIBCO iProcess Objects Director that responds to the UDP broadcast. By default, TIBCO iProcess Objects Directors listen for UDP broadcasts on UDP port 28001. You can change this for a TIBCO iProcess Objects Director using the UDP_SERVICE_NAME process attribute (see [UDP_SERVICE_NAME](#)).

For more information about UDP broadcasts, see *TIBCO iProcess Server Objects Programmer's Guide*.

- **Directed UDP Message** The client can issue a directed UDP message to a TIBCO iProcess Objects Director by calling the verifyNode method on sNodeManager. If TIBCO iProcess Objects Director to which the UDP message was directed responds, the method call returns a vNode object that represents that TIBCO iProcess Objects Director. By default, TIBCO iProcess Objects Directors listen for UDP messages on UDP port 28001. You can change this for a TIBCO iProcess Objects Director using the UDP_SERVICE_NAME process attribute (see [UDP_SERVICE_NAME](#)).

For more information about directed UDP messages, see *TIBCO iProcess Server Objects Programmer's Guide*.

- **Manually Creating a vNodeId Object** If you have all of the required information (node name, computer name, IP address, and TCP port), you can construct a vNodeId object that represents TIBCO iProcess Objects Director. The alsDirector parameter in the vNodeId constructor must also be set to True to indicate that the object represents TIBCO iProcess Objects Director.

For more information about manually creating an SWNodeInfo object, see *TIBCO iProcess Server Objects Programmer's Guide*.

Connecting to a TIBCO iProcess Objects Server via a Director

How you connect to a TIBCO iProcess Objects Server via a TIBCO iProcess Objects Director depends on whether you are using TIBCO iProcess Objects or TIBCO iProcess Server Objects, as follows:

TIBCO iProcess Objects

The key of the SWNodeInfo object (available in the Key property) obtained through one of the methods described in the previous subsection identifies whether the object represents a TIBCO iProcess Objects Server or Director, as follows:

```
SWNodeInfo.Key = ComputerName|NodeName|IsDirector|InstanceNumber
```

If *IsDirector* = Y, it represents a Director; if *IsDirector* = N, it represents a Server.

Once the client has an *SWNodeInfo* object that represents a TIBCO iProcess Objects Director, the key from that object can be passed in the *NodeKeys* parameter of the *Login* method:

```
Login (NodeKeys, Password, [UserName])
```

If the *NodeKey* represents a Director (*IsDirector* = Y), the Director uses the “pick method” specified when the Director was configured to determine which TIBCO iProcess Objects Server the client should connect to. Internally, a TCP connection is established between the client and the TIBCO iProcess Objects Server that the Director selected. From the user’s standpoint, the selection and connection to the TIBCO iProcess Objects Server is transparent. All future transmissions, while that user is logged in, are made directly to the TIBCO iProcess Objects Server instance.

Note that if the client fails to connect to the TIBCO iProcess Objects Server (for example, an invalid user name or password is passed), it will NOT be logged in the TIBCO iProcess Objects Director’s log. Once TIBCO iProcess Objects Director gives the client a TIBCO iProcess Objects Server to log in to, communication with TIBCO iProcess Objects Director is complete. You must use the TIBCO iProcess Objects Server’s log to troubleshoot this type of problem.



Note

For efficiency reasons, after having TIBCO iProcess Objects Director choose a TIBCO iProcess Objects Server for you, you should log in directly to that same TIBCO iProcess Objects Server for all subsequent logins. This is because a SAL session is started for a user the first time they log in; subsequent logins can use that same SAL session, resulting in a much more efficient and faster login.

TIBCO iProcess Server Objects

To connect to a TIBCO iProcess Objects Server via a TIBCO iProcess Objects Director, you must have a *vNodeId* object that represents the Director (the *vNode* object returned by *getNodes* or *verifyNode* can be cast to a *vNodeId* object).

Pass the *vNodeId* object that represents TIBCO iProcess Objects Director in the constructor for the desired Server Object. For example:


```
sUser(vNodeId aNodeId,  
      String aUserName,  
      String aPassword)
```

If the `vNodeId` object represents TIBCO iProcess Objects Director, the Director use the “pick method” specified when the Director was configured to determine which TIBCO iProcess Objects Server the Server Object should connect to. Internally, a TCP connection is established between the Server Object and the TIBCO iProcess Objects Server that the Director selected. From the user’s standpoint, the selection and connection to the TIBCO iProcess Objects Server is transparent. All future transmissions, while that user is logged in, are made directly to the Server instance.

Note that if the client fails to connect to the TIBCO iProcess Objects Server (for example, an invalid user name or password is passed), it will NOT be logged in the TIBCO iProcess Objects Director’s log. Once the TIBCO iProcess Objects Director gives the Server Object a TIBCO iProcess Objects Server to log in to, communication with TIBCO iProcess Objects Director is complete. You must use the TIBCO iProcess Objects Server log to troubleshoot this type of problem.

**Note**

For efficiency reasons, after having the TIBCO iProcess Objects Director choose a TIBCO iProcess Objects Server for you, you should log in directly to that same TIBCO iProcess Objects Server for all subsequent logins. This is because a SAL session is started for a user the first time they log in; subsequent logins can use that same SAL session, resulting in a much more efficient and faster login. (Use the `getNodeId` method to get the `vNodeId` object for the particular instance of the TIBCO iProcess Objects Server to which you are connected so you can connect to that same instance at a later time.)

Director-Related Properties and Methods

The following subsections described the Director-related properties and methods for TIBCO iProcess Objects and TIBCO iProcess Server Objects.

TIBCO iProcess Objects

An `SWNodeInfo` object that represents TIBCO iProcess Objects Director contains properties and methods that provide information about that instance of the Director.

- **ClientCnt** The number of TIBCO iProcess Objects Servers that the Director knew about when the SWNodeInfo object was obtained.
- **ClusterId** This property is used to uniquely identify the node to which the Director is associated. It consists of the "client/server" RPC port number found in the engines' swdefs file and the machine name on which the database is installed.
- **ComputerName** The name of the machine on which TIBCO iProcess Objects Director is installed.
- **DirectorNodeInfos** Returns a list of SWNodeInfo objects, one for each TIBCO iProcess Objects Server the Director knows about. This list is updated dynamically as TIBCO iProcess Objects Servers are added to or removed from the process_config table.

Note: If there is an SWNodeInfo object in this list that represents a TIBCO iProcess Objects Server that has a status other than swAvailable, the following properties on that SWNodeInfo object returns an empty string: SWEOSrvVersion, SWEOSrvName, SWEOSrvDesc, SWVersion, and ClusterId.
- **InstanceNumber** The instance number of TIBCO iProcess Objects Director.
- **IsDirector** This identifies whether the SWNodeInfo object represents a TIBCO iProcess Objects Server or Director. True = Director; False = Server.
- **Key** A string that identifies the specific TIBCO iProcess Objects Director. This key can be passed in the *NodeKeys* parameter of the Login method to cause the Director to select a Server.
- **Status** This returns the current status of TIBCO iProcess Objects Director. This value is enumerated using the SWNodeInfoStatusType enumeration (for example, swAvailable, swNotRunning, etc.)
- **SWEOSrvDesc** Defaults to "TIBCO iProcess Objects Director". This is set with the UDP_SERVICE_DESC process attribute (see [UDP_SERVICE_NAME](#)).
- **SWEOSrvName** Defaults to "SPODirector."

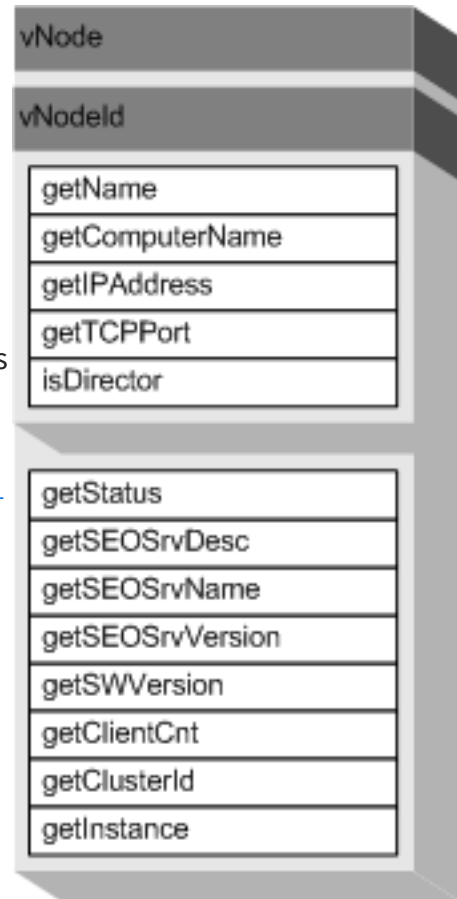
SWNodeInfo	
<	ClassId
<	ClientCnt
<	ClusterId
<	ComputerName
<	DirectorNodeInfos L
<	InstanceNumber
<	IPAddr
<	IsDirector
<	Key
<	Name
<	Status
<	SWEOSrvDesc
<	SWEOSrvName
<	SWEOSrvVersion
<	SWVersion
<	Tag
<	TCPPort
* Interface	
* InterfaceEqual	
* InterfaceNewer	

- **SWEOSrvVersion** The version number of TIBCO iProcess Objects Director.
- **SWVersion** The version of TIBCO iProcess Engine.
- **TCPPort** This identifies the TCP port on which TIBCO iProcess Objects Director communicates with clients. This port is established on the Director using the TCP_SERVICE_NAME process attribute (see [TCP_SERVICE_NAME](#)).
- **Interface** The interface version number of TIBCO iProcess Objects Director. Note that this number is generally not used to determine compatibility between the client and TIBCO iProcess Objects Director – compatibility is determined by TIBCO iProcess Objects Director when the client attempts to connect to TIBCO iProcess Objects Director.
- **InterfaceEqual** This method allows you to determine whether or not TIBCO iProcess Objects Director's interface version number is equal to the value passed in the method call.
- **InterfaceNewer** This method allows you to determine whether or not TIBCO iProcess Objects Director's interface version number is newer (greater) than the value passed in the method call.

TIBCO iProcess Server Objects

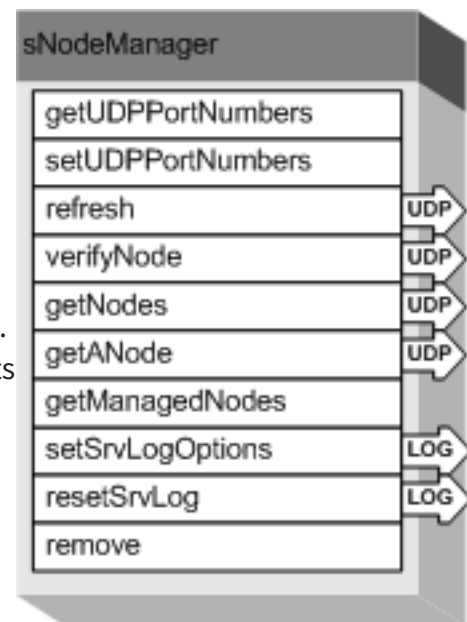
The following describes the methods on the vNode and vNodeId objects that provide information about TIBCO iProcess Objects Director when those objects represent TIBCO iProcess Objects Director.

- **getName** The name of the node the Director is associated with.
- **getComputerName** The name of the machine on which TIBCO iProcess Objects Director is installed.
- **getIPAddress** The IP address of the machine on which TIBCO iProcess Objects Director is installed.
- **getTCPPort** The TCP port on which TIBCO iProcess Objects Director communicates with clients. This port is established on the Director using the TCP_SERVICE_NAME process attribute (see [TCP_SERVICE_NAME](#)).
- **isDirector** This identifies whether the vNodeid object represents a TIBCO iProcess Objects Server or a TIBCO iProcess Objects Director. True = Director; False = Server.
- **getStatus** The current status of TIBCO iProcess Objects Director. This value is enumerated using the SWNodeInfoStatusType enumeration (for example, swAvailable, swNotRunning, etc.)
- **getSEOSrvDesc** Defaults to “TIBCO iProcess Objects Director”. This is set with the UDP_SERVICE_DESC process attribute (see [UDP_SERVICE_DESC](#)).
- **getSEOSrvName** Defaults to “SPODirector.”
- **getSEOSrvVersion** The version number of TIBCO iProcess Objects Director.
- **getSWVersion** The version of TIBCO iProcess Engine.
- **getClientCnt** The number of TIBCO iProcess Objects Servers that the Director knew about when the vNode object was obtained.
- **getClusterId** This method is used to uniquely identify the node to which the Director is associated. It consists of the "client/server" RPC port number found in the engines' swdefs file and the machine name on which the database is installed.
- **getInstance** The instance number of TIBCO iProcess Objects Director.



The sNodeManager object also contains a method that allows you to determine the TIBCO iProcess Objects Servers that TIBCO iProcess Objects Director knows about.

The getManagedNodes method returns an array of vNodeId objects, one for each TIBCO iProcess Objects Server that TIBCO iProcess Objects Director knows about. This list is updated dynamically as TIBCO iProcess Objects Servers are added to or removed from the process_config table.



Pageable Lists/SWXLists of Work Items

If you are using a TIBCO iProcess Objects Director to connect to a TIBCO iProcess Objects Server, be aware that a pageable list (TIBCO iProcess Server Objects) or SWXList (TIBCO iProcess Objects) of work items or predicted work items is tied to a specific instance of the TIBCO iProcess Objects Server. If a pageable list/SWXList of work items or predicted work items is created, that list can only be accessed on the specific instance of the TIBCO iProcess Objects Server where it was created. This is not just limited to getting the work items on the pageable list, but also to the method calls on work items obtained from the pageable list/SWXList. This is because the list holds the state to the Work Item Server.

TIBCO iProcess Objects Director Logging

The following logs are used by TIBCO iProcess Objects Director:

- **TIBCO iProcess Objects Director Log** Process attributes are used to specify the type and amount of information to write to this log. For more information, see [TIBCO iProcess Objects Director Log](#).
- **UNIX System Log (UNIX systems only)** Errors and other operational information generated by TIBCO iProcess Objects Director are written to this standard UNIX log. This log also contains errors/information messages from other UNIX facilities and programs running on the box. For more information, see [UNIX System Log](#).

- **Windows Event Log** (Windows systems only) Errors and other operational information generated by TIBCO iProcess Objects Director are written to this standard Windows log. This log also contains errors/information messages from other Windows processes and programs running on the box.

You can control whether or not messages are written to this log using the `WRITEERRSTOEVENTLOG` process attribute — see [WRITEERRSTOEVENTLOG](#).

- **dir_error File (UNIX systems only)** This TIBCO iProcess Objects Director-specific error file is created when the Director is installed. It contains error messages generated by TIBCO iProcess Objects Director. It is located in the `$SWDIR/logs` directory (where `$SWDIR` is the directory in which TIBCO iProcess Engine is installed).

TIBCO iProcess Objects Director Log

The first time TIBCO iProcess Objects Director is used, it creates a log file that records messages generated by TIBCO iProcess Objects Director. The name and location of the TIBCO iProcess Objects Director log file are as follows:

- **Windows** `SWDIR\logs\spodirectorXX.log`
- **UNIX** `$SWDIR/logs/spodirectorXX.log`

where `XX` is the instance number of TIBCO iProcess Objects Director. If only a single Director process is running, it is “01”.

Note: To change the log files directory, specify the directory in the `staffpms` file located in the `SWDIR/etc` directory. For more information, see "Configuring Log Files Directory" in *TIBCO iProcess Engine Administrator's Guide*

The amount and type of data written to the TIBCO iProcess Objects Director log is controlled by the following process attributes:

- `LOG_LEVEL` level of information written to the log.
- `LOG_CATEGORIES` the categories of information to write to the log.
- `LOG_FILE_MAX_SIZE` the maximum size of the log file.
- `LOG_FILE_MAX_ARCHIVES` the number of archive log files to create if the log rolls over. See the [Archived TIBCO iProcess Objects Director Log Files](#).

- `TRACE_MSG` flag specifying whether network (request/response) messages are written to the log. Note that request/response messages are written to the log if the `LOG_LEVEL` attribute is set to 4 (Debug), regardless of the setting of this attribute.

These are described in more detail in the table on the preceding pages.

The TIBCO iProcess Objects Director log is formatted in the same way as the TIBCO iProcess Objects Server log — for information about this format, see *TIBCO iProcess Objects Server Administrator's Guide*.

Archived TIBCO iProcess Objects Director Log Files

If the TIBCO iProcess Objects Director log file reaches the maximum size specified by the `LOG_FILE_MAX_SIZE` process attribute, the log is rolled over. If desired, you can specify that when the log rolls over, the previous log file is archived. This is specified using the `LOG_FILE_MAX_ARCHIVES` process attribute. The default is to not save archived log files.

Archiving log files can be useful if you want to collect many MB of debug log, but you do not want to deal with a very large log file by setting `LOG_FILE_MAX_SIZE` to a large value.

Archived log files are named `spodirectorXX_archive_n.log`, where `XX` is the instance number of TIBCO iProcess Objects Director, and `n` is a counter that is incremented each time the log rolls over (starting at 1). For example, if the `LOG_FILE_MAX_ARCHIVES` process attribute is set to 2, the first time the log rolls over, it is saved as `spodirectorXX_archive_1.log`. The next time it rolls over, it is saved as `spodirectorXX_archive_2.log`. If it rolls over again, it is saved as `spodirectorXX_archive_3.log`, but `spodirectorXX_archive_1.log` will be deleted because it is only saving two archives.

UNIX System Log

When running TIBCO iProcess Objects Director in a UNIX environment, error messages can be written to the standard UNIX system log (other UNIX facilities and programs also write errors/information messages to this log). You can control whether or not messages are written to this log using the `WRITEERRSTOEVENTLOG` process attribute — see [WRITEERRSTOEVENTLOG](#).

The location of the UNIX system log can be configured on each UNIX system, but the usual location is `/var/log/messages` on Linux.

All syslog messages are categorized by the type of “subsystem” or “facility” that originated the message, and by the “priority” given the message. The “subsystems” are areas such as “kernel” (message generated by the kernel, i.e., UNIX itself), “user” (messages from various user programs), “mail,” “daemon,” “auth,” and “lpr.” Additionally, “local” subsystems

(local0 through local7) are reserved for local program use. The TIBCO iProcess Objects Director uses one of these — local0.

Within each subsystem, various priority levels exist. In TIBCO iProcess Objects Director, the priorities that are used correspond to the debug log levels/types. They are:

- **local0.info** includes “info,” “notice,” “warn,” and “err”
- **local0.notice** includes “notice,” “warn,” and “err”
- **local0.warn** includes “warn,” and “err”
- **local0.err** includes “err” only

Notice that each priority also includes the levels it.

The UNIX system log file is controlled by the configuration file `/etc/syslog.conf`. You could optionally choose to send all TIBCO iProcess Objects Director messages to a different file by adding a line similar to the following to the `syslog.conf` file:

```
local0.info /var/adm/spo_director_messages_only
```

Note that whenever the `syslog.conf` file is changed, the `syslogd` daemon must be sent a `SIGHUP` signal. For example:

```
kill -HUP `cat /etc/syslog.pid`
```


TIBCO Documentation and Support Services

For information about this product, you can read the documentation, contact TIBCO Support, and join TIBCO Community.

How to Access TIBCO Documentation

Documentation for TIBCO products is available on the [Product Documentation website](#), mainly in HTML and PDF formats.

The [Product Documentation website](#) is updated frequently and is more current than any other documentation included with the product.

Product-Specific Documentation

The documentation for this product is available on the [TIBCO iProcess® Engine Product Documentation](#) page.

How to Contact Support for TIBCO Products

You can contact the Support team in the following ways:

- To access the Support Knowledge Base and getting personalized content about products you are interested in, visit our [product Support website](#).
- To create a Support case, you must have a valid maintenance or support contract with a Cloud Software Group entity. You also need a username and password to log in to the [product Support website](#). If you do not have a username, you can request one by clicking **Register** on the website.

How to Join TIBCO Community

TIBCO Community is the official channel for TIBCO customers, partners, and employee subject matter experts to share and access their collective experience. TIBCO Community offers access to Q&A forums, product wikis, and best practices. It also offers access to extensions, adapters, solution accelerators, and tools that extend and enable customers to gain full value from TIBCO products. In addition, users can submit and vote on feature

requests from within the [TIBCO Ideas Portal](#). For a free registration, go to [TIBCO Community](#).

Legal and Third-Party Notices

SOME CLOUD SOFTWARE GROUP, INC. (“CLOUD SG”) SOFTWARE AND CLOUD SERVICES EMBED, BUNDLE, OR OTHERWISE INCLUDE OTHER SOFTWARE, INCLUDING OTHER CLOUD SG SOFTWARE (COLLECTIVELY, “INCLUDED SOFTWARE”). USE OF INCLUDED SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED CLOUD SG SOFTWARE AND/OR CLOUD SERVICES. THE INCLUDED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER CLOUD SG SOFTWARE AND/OR CLOUD SERVICES OR FOR ANY OTHER PURPOSE.

USE OF CLOUD SG SOFTWARE AND CLOUD SERVICES IS SUBJECT TO THE TERMS AND CONDITIONS OF AN AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER AGREEMENT WHICH IS DISPLAYED WHEN ACCESSING, DOWNLOADING, OR INSTALLING THE SOFTWARE OR CLOUD SERVICES (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH LICENSE AGREEMENT OR CLICKWRAP END USER AGREEMENT, THE LICENSE(S) LOCATED IN THE “LICENSE” FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE SAME TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

This document is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of Cloud Software Group, Inc.

TIBCO, the TIBCO logo, the TIBCO O logo, ActiveMatrix BusinessWorks, Business Studio, Enterprise Message Service, Hawk, iProcess, and Rendezvous are either registered trademarks or trademarks of Cloud Software Group, Inc. in the United States and/or other countries.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only. You acknowledge that all rights to these third party marks are the exclusive property of their respective owners. Please refer to Cloud SG’s Third Party Trademark Notices (<https://www.cloud.com/legal>) for more information.

This document includes fonts that are licensed under the SIL Open Font License, Version 1.1, which is available at: <https://scripts.sil.org/OFL>

Copyright (c) Paul D. Hunt, with Reserved Font Name Source Sans Pro and Source Code Pro.

Cloud SG software may be available on multiple operating systems. However, not all operating system platforms for a specific software version are released at the same time. See the “readme” file for the availability of a specific version of Cloud SG software on a specific operating system platform.

THIS DOCUMENT IS PROVIDED “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. CLOUD SG MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S), THE PROGRAM(S), AND/OR THE SERVICES DESCRIBED IN THIS DOCUMENT AT ANY TIME WITHOUT NOTICE.

THE CONTENTS OF THIS DOCUMENT MAY BE MODIFIED AND/OR QUALIFIED, DIRECTLY OR INDIRECTLY, BY OTHER DOCUMENTATION WHICH ACCOMPANIES THIS SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY RELEASE NOTES AND "README" FILES.

This and other products of Cloud SG may be covered by registered patents. For details, please refer to the Virtual Patent Marking document located at <https://www.tibco.com/patents>.

Copyright © 1994-2024. Cloud Software Group, Inc. All Rights Reserved.