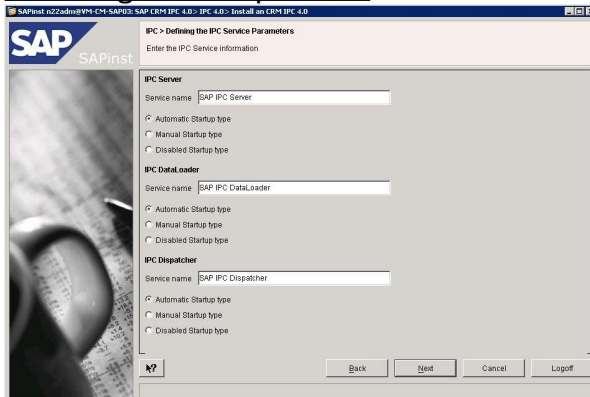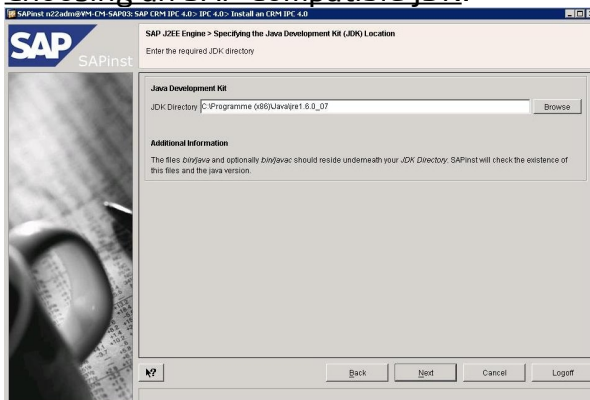# Installation & Setup of the IPC

## Installation

The IPC is a Java based application suite, consisting of a *Dispatcher*, *Server* and *Monitoring Tools*:

- An SAP compatible JDK must have been installed as a prerequisite: *http://www.sun.com/software/javaforbusiness/sap_download.jsp*
- Installing IPC Components:



- Choosing an SAP Compatible JDK:



## Setup:

To set up IPC as an RFC server and connect it to a CRM system, RFC destinations (one for the dispatcher and one for every IPC server connected to it) must be created, and a few properties must be maintained in file "*dispatcher.properties*".

## 1st Step - Preparations:

- The decision about how many IPC Servers to configure must be made when installing the IPC.
  Because, other than in the socket-based scenario, one can not dynamically add IPC Servers to a running Dispatcher.
  Here a decision must be made on how many servers are appropriate, CRM and the Dispatcher be configured accordingly, and finally the Dispatcher must be started.
- Example for installing 2 IPC servers -
  Since CRM needs to talk to the dispatcher to get connected to a(ny of the) server(s), three destinations must be maintained.
- **CAUTION:** As a convention, RFC destinations and RFC program IDs are the same. Additionally, all server destinations are similar - they start with the same string, followed by a number.
  Only the dispatcher's destination and the "prefix string" of the servers' destinations can be choosen.

| RFC Destination | Explanation |
|---|---|
| IPC_RFC_DISPATCHER | RFC destination (and program ID) of the dispatcher. |
| IPC_RFC_SERVER_ | Prefix string for the servers.<br>This results in RFC destinations/program IDs *IPC_RFC_SERVER_1* and *IPC_RFC_SERVER_2* for the servers. |

## 2nd Step - Setting up IPC RFC-Destinations in SAP:

- Log into the SAP System from which the IPC is to be accessed.
  In this system, maintain the RFC destinations in a similar way as *SAPCRM_SPC* et al. - ie.
  maintain an entry in table *SMOFPARSFA* where the key is e.g. *IPC_RFC_DISPATCHER*
  and parval1 is *IPC_RFC_DISPATCHER* (this step is actually optional, as it gives a client-
  dependent level of indirection that might help in some scenarios).
- List of IPC RFC-Destinations:
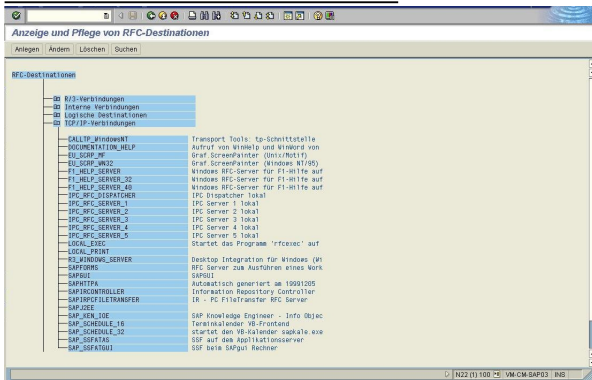


- Then configure each destination, as as real RFC destination of (*Type "T"*).
- Enter an appropriate SAP gateway and the corresponding destination name.
  **CAUTION:** At this time the IPC is not yet configured properly, thus testing the
  destination now will result in failure!

<u>3rd Step - Setting-up and starting the IPC</u>:

- Edit file "*<InstallDir>/lib/properties/dispatcher.properties*" (may not yet exist if empty).
- There enter the following properties:
  - rfcEnabled = true
  - rfcDispatcherProgId = IPC_RFC_DISPATCHER
  - rfcServerCount = 2
  - rfcServerProgIdPrefix = IPC_RFC_SERVER_
- When the dispatcher starts up, it needs to contact a SAP system to find out more RFC settings (e.g. the RFC Gateway to be used).
  If using a connection to a SAP system for DB access, then this connection is automatically used by the dispatcher to retrieve this info.
  Only if using a local database, the following properties have to be entered into "dispatcher.properties":
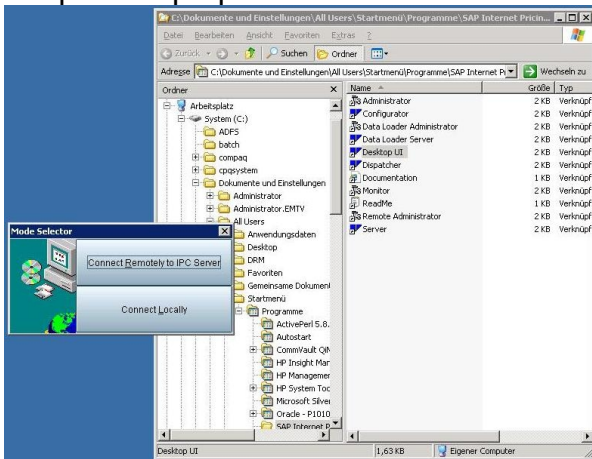


- If these five properties have been maintained, they will always be used for the calls to the SAP system:

  rfcSapSystemUser = <User ID with RFC access in SAP System>
  rfcSapSystemPassword = <User's password, in plain text>
  rfcSapSystemSapClient = <SAP client to use>
  rfcSapSystemApplicationServer = <Host-ID of SAP Application Server>
  rfcSapSystemSystemNumber = <SAP System Number>

  Only if they have not been maintained, the default DB connection will be used.
  It is recommended to leave these settings alone and use the DB connection instead.
  This can be set up in the administrator (verify settings), and the password is then stored in an encoded way.
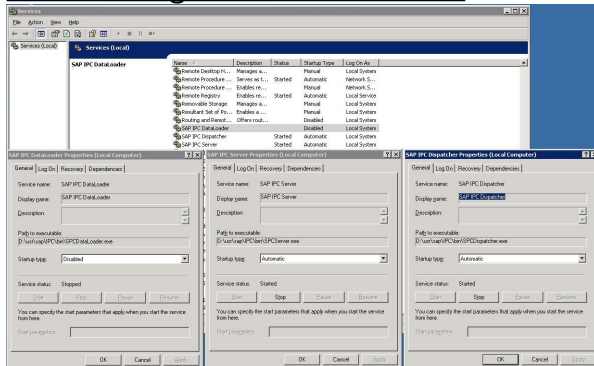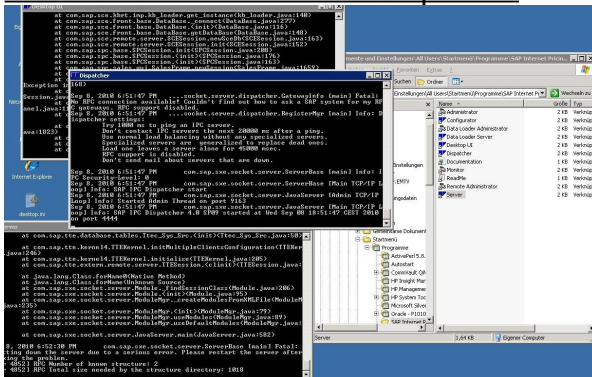
- <u>Local Maintenance Administrator</u>:
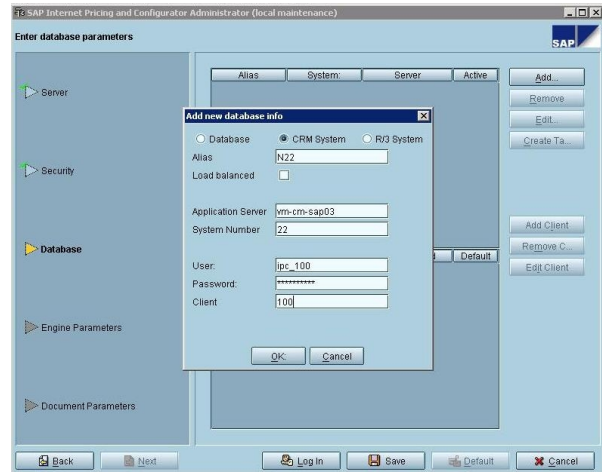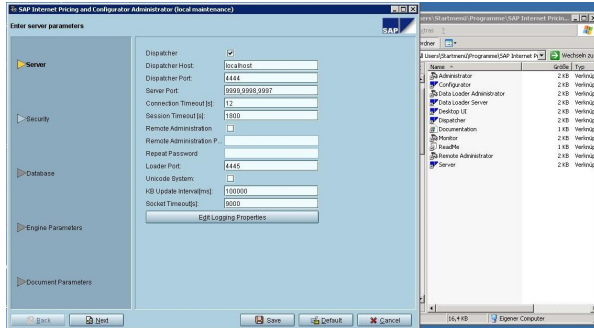


- Set up the rest of IPC as described in the installation guide and start the IPC services (which were registered by the install program).
  Once this is done, the dispatcher and its servers have already registered at their RFC destinations.
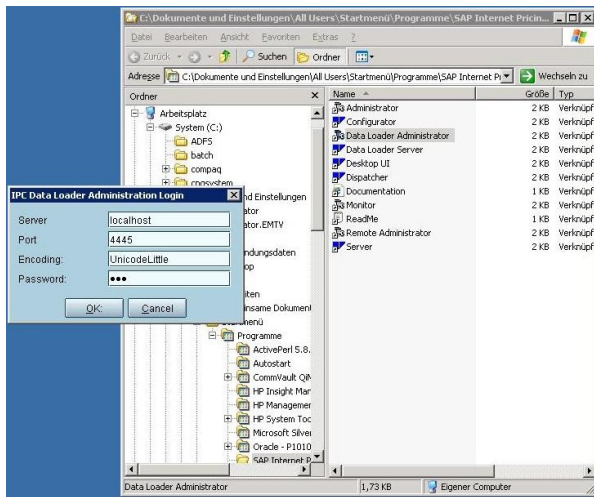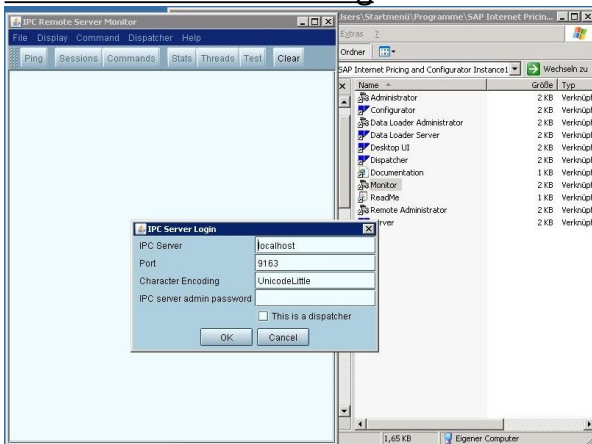


- <u>Start the registered IPC-Services</u>:

- <u>Don't start IPC via the Desktop Icons!</u>



**CAUTION:** Starting the Dispatcher and/or the Server via the IPC Desktop icons will not result in a functioning setup, as neither the RFC registration will succeed this way, nor will the IPC be available after the Windows user session has terminated.

- <u>Remote Server Monitoring</u>:



- To completely bypass the process by which the dispatcher retrieves its gateway settings and if to limit the setup to using a single RFC gateway for the IPC dispatcher and all of its servers, the gateway host and port # can be directly specified in file "dispatcher.properties":
  *rfcGateways = my.gateway.host:myGatewayPort*
  With this setting there is no need to maintain property settings for an SAP system for DB access, and no need to specify the *rfcSapSystem\** properties.
  All IPC components will register at the specified gateway host & port, so the corresponding entries must have been maintained in TA "SM59" at the SAP target system.


<u>4th Step - Destination Testing</u>:

- TA "SM37" ☐ COM_IPC_SESSION_BEGIN ☐ ...
- The report will
  • Connect to the IPC-Dispatcher and request it to return the Server info.
  • Connect to the IPC-Server to create a new server session on it.
- If everything is set up correctly, a "result screen" with a number to the right of "EV_SESSION_ID" at the bottom of the table will be displayed.
  Otherwise an ABAP short dump might be created.

## Adding New Commands:

- After adding new commands to the IPC-Server or changing parameters of existing IPC commands, RFC clients need to know about this change.
  The repository info about all IPC commands that are available as RFC functions is maintained in a central XML File in: *<InstallDir>/lib/properties/RFCCommands.xml*.
  If the bridge tool has been used (⇒ IPC v2.0C), this file is nearly identical to *SpcCommands.xml*.
- For each command a single XML tag needs to be maintained - this tag defines the import, table and export parameters.
  Parameter definitions include:
  • An RFC and a Java name for the parameter.
  The Java name is the actual name of the corresponding IPC parameter, the RFC name is the name under which you want to access it from ABAP.
  • The size of the field in ABAP/4.
  • Flags if the command should be broadcasted to all IPC servers and if the value is a normal string or a UTF8 encoded unicode string.
- There are a lot of command definitions in the shipped version of file *RFCCommands.xml*. They can be copied and changed according to customer needs.
  Just ensure that the parameter lengths match those in ABAP/4, as mismatch of parameter lengths will cause bizarre errors because the JCO inside of IPC will misinterpret interpret the data passed to it (e.g. it will understand "*ABC*", "*DEF*" as "*BCD*", "*DE<garbage>*").

## IPC Connectivty with the SAP CRM System can be tested via:

- TA "SM59":
  *TCP/IP Connections -> [IPC_RFC_Dispatcher¦IPC_RFC_SERVER_[1-5]] -> Button: "Test Connection".*
- TA "SE37":
  *FunctionModule: "COM_IPC_SESSION_BEGIN" -> Menu: "Function Module" -> Test -> Single Test.*