TA system configuration and tacon.properties

The Webdesk TA Connector (Interflex IF6020) is configured through the file tacon.properties. The file defines which technical user the connector uses to log in to the IF6020, which IF6020 accounts and fields are used for vacation, working time and persons, and how bookings, the attendance board and the calendar behave. Most values can be edited in Webdesk in the mask "Systemkonfiguration" (action ta_editSystemConfig) without accessing the connector server. This page is intended for administrators and consultants who set up or maintain a TA connector. An installed and running Webdesk TA Connector is a prerequisite (installation: Webdesk TA Connector[1]); Webdesk system parameters of the time and attendance module are described on Systemparameter TA[2].

How Webdesk, the connector and the file work together

- **Location:** The file is located in the conf directory of the TACon installation (e.g. TACon\conf). The connector finds it on the classpath (wrapper.conf contains ../conf). The connector does not start without the file. Webdesk itself never reads the file directly but asks the connector via RMI.
- **Connector start:** The broker (TaBrokerStart) first renames obsolete keys (see section "Automatic adjustments at startup"). Afterwards every entry of the form beanname.property is applied as an override to the bean of that name, both in the broker and in every RMI server. Keys for which no bean or property exists are ignored without an error message, so typos in a key go unnoticed.
- **Hand-over to the RMI servers:** After an RMI server has started, the broker reads the keys that can be edited in the mask from the file and passes them to the server. These values take precedence over the defaults built into the software. A missing key counts as empty or false and overrides the internal default. All keys of the mask should therefore exist in the file; saving in the mask writes all of them automatically.
- **Editing in the mask:** When the mask is opened, Webdesk fetches the current configuration from the broker via RMI (address in webdesk.properties, see below). On saving, the broker writes all values to tacon.properties and passes them immediately to all running RMI servers. A restart is only necessary for the exceptions noted in the key reference. With a clustered connection (several addresses separated by ; in webdesk.ta.if6020impl.rmi) Webdesk forces a failover after saving and saves the configuration on the second connector as well.
- **Editing the file by hand:** The broker keeps the file in memory. Manual changes only take effect after a restart of the service "Webdesk TA Connector for IF6020". If someone saves in the mask beforehand, the file content is overwritten with the broker's old state. Therefore edit the file manually only while the service is stopped.

Mask "Systemkonfiguration"

You find the mask in the menu under Time and Attendance > IF6020 > Systemkonfiguration (action ta_editSystemConfig). It always shows exactly one configuration (that of the connector) and cannot be deleted. The group headings of the mask are fixed in English and are not translated. The fields of the mask map to the keys of the file as follows:

Group in the mask	Keys / content
Authentication	Credentials for the IF6020 (TaRmiService.*, credentials.encrypted)
Holiday Display (planned/unplanned)	Vacation accounts and vacation reference date (TaJournalDAO.*), also the field "Use double reaccounting on account correction"

Person Settings	Person fields and person management (TaPersonDAO.*, TaPersonDAOImpl.*, TaVacationEntitlementDAO.taPersonFieldEntryDate)
Vacation Calculation Binding	Vacation entitlement and levels (TaVacationEntitlementDAO.*)
Booking Parameters	Booking behavior (TaBookingDAO.*), the two lists of absence reason IDs (read-only), "Booking messages are returned as correction comments"
Accounting Parameters	Working time accounts and maximum working times (TaAccountDAO.*, TaBrokerImpl.personTaIdsReservedForCalculations)
Calendar Configuration	Calendar and absences (TaCalendarDAO.*, TaAbsenceIntervalDAO.*)
StatusList Configuration	Attendance board (TaStatusTableauDAO.*)
Timemodel Administration	Number ranges for new daily and weekly programs (TaTimeModelDAO.minId, TaTimeModelProgramDAO.minId)
TA Data Warehouse	Switch for the data warehouse (TaDataWarehouseDAO.warehouseEnabled)

- Read-only (no input field): "Credentials encrypted", "Name of field storing PoPerson UID", "Name of the field storing the employee ID", "Name of the field storing the effective employment marker", "IDs of productive absence reasons" and "IDs of absence reasons with attendant behavior". You change these values only in the file (with the service stopped) or via the absence reason administration.
- Fields for IF6020 accounts, person fields, daily and weekly program fields and absence reasons are filled as selection lists from the IF6020.
- Keys that do not appear in the mask (see "Further keys, file only") are kept unchanged in the file when saving.

Credentials and encryption

The connector logs in to the IF6020 with a technical user. User name and password are stored in tacon.properties and inserted as placeholders into the RMI servers when the connector starts. If the credentials are encrypted, the connector decrypts them at startup with the key encryption.key from the same file.

Encryption procedure: When Webdesk starts, it passes its encryption key to the connector. If encryption.key is missing in the file and credentials.encrypted is not explicitly set to false, the connector encrypts user name and password with this key, writes the key (Base64) and sets credentials.encrypted=true. If credentials.encrypted=false is set, the credentials stay in plain text. In the same step Webdesk writes the two lists of absence reason IDs if they are still empty in the file (see section absence reason lists).

Warning: Never pass on the value of encryption.key or store it in documentation, tickets or version control. If the key is lost, the encrypted credentials can no longer be read. For backing up the Webdesk key see Neue Features in 4.56[3].

Note: With encrypted credentials the mask shows a button for changing instead of the fields "User" and "Password" (e.g. "Change user"). New values are encrypted with the key from the file when saving. After changing user or password the service "Webdesk TA Connector for IF6020" must be restarted.

Property	Description	Possible value(s)
TaRmiService.username	**Required:** yes. Technical user in the IF6020 the connector connects with. Mask: "User". With encrypted credentials this holds the encrypted Base64 value. A change takes effect only after a restart of the connector.	text (default: no default; the shipped file contains an example user that must be replaced with the installation's user)
TaRmiService.password	**Required:** yes. Password of the technical user. Mask: "Password". With encrypted credentials this holds the encrypted Base64 value. A change takes effect only after a restart of the connector.	text (default: no default; the shipped file contains an example password in plain text that must be replaced)
credentials.encrypted	**Required:** no. Marks whether user name and password are encrypted. Mask: "Credentials encrypted" (read-only). Set at the first Webdesk start (see above). If the key is missing, the connector reads the credentials as plain text.	true/false (default: not set = plain text)
encryption.key	**Required:** no, entered automatically. Base64 key with which the credentials are encrypted. Corresponds to the encryption key of Webdesk. Not visible in the mask. Never pass the value on.	Base64 text (default: not set)

Key reference

For each key the tables give the effect including whether the key is required, the corresponding field of the mask (in quotation marks, as on screen) and the default value. "Default" is the value that applies when the key is missing; "shipped file" means the tacon.properties of the installation package. "Required: yes" means that the connector does not work, or not meaningfully, without a value; "conditional" applies only to the named function; "no" means optional. Account and field names are IF6020 names (e.g. vkkonto4, stpinfo120, stwert5). Lists are comma-separated unless stated otherwise. Only keys that the code actually evaluates are listed.

Connection, broker and queue

The Webdesk side of the connection is defined in webdesk.properties; the connector service is started through wrapper.conf. Neither are entries of tacon.properties, but they are listed here because they are checked together during setup. The keys in the table can only be changed in the file (not in the mask).

Property	Description	Possible value(s)
TaBrokerQueueHandler.startNewRmiServerIfCrashed	**Required:** no. If true, the broker starts a new RMI server when a server has crashed or no working server is available (as long as the permitted number is not exceeded). If false, an error is	true/false (default: false)

	reported in that case. **Note:** The shipped file contains the line TaBrokerQueue.startNewRmiServerIfCrashed=true. In the code the bean is called TaBrokerQueueHandler; a key with the prefix TaBrokerQueue finds no bean and has no effect.	
TaBrokerQueue.portReuseQuarantineMinutes	**Required:** no. Only when running with the object pool (optional start parameter useObjectPool): time in minutes during which the port of a failed or removed RMI server is not assigned again. Missing, invalid or non-positive values are replaced by the default.	integer in minutes (default: 30)
TaBroker.statusTableauFanOut.maxServers	**Required:** no. Only when running with the object pool: maximum number of RMI servers over which one attendance board request may be distributed.	integer (default: 4)
TaBroker.statusTableauFanOut.reserveServers	**Required:** no. Only when running with the object pool: number of RMI servers that stay free for other calls (above all bookings) when an attendance board request is distributed.	integer (default: 2)
TaBroker.statusTableauFanOut.minPersonsPerServer	**Required:** no. Only when running with the object pool: minimum number of persons per RMI server when distributing. Smaller requests are not split.	integer (default: 25)

Counterpart and start parameters outside tacon.properties

Property	Description	Possible value(s)
webdesk.properties: webdesk.ta.if6020impl.rmi	Address of the broker as seen from Webdesk, e.g. rmi://localhost:4009/taRmiBroker. With several connectors (cluster) the addresses are separated by ;.	text (installation default: rmi://localhost:4009/taRmiBroker)
webdesk.properties: webdesk.ta.if6020impl.rmi.timeoutMillis	Time limit in milliseconds for connection handshake to the broker.	number (default: 2000)
webdesk.properties: webdesk.ta.if6020impl.rmi.connectTimeoutMillis	Time limit in milliseconds for establishing the TCP connection to the broker.	number (default: 3000)
wrapper.conf: wrapper.app.parameter.2	Port of the broker (and start port; the RMI servers use the following ports).	number (installation default: 4009)

wrapper.conf: wrapper.app.parameter.3	Number of RMI servers running concurrently, i.e. of parallel possible accesses to the IF6020. An update keeps the previous value.	number (default of the shipped wrapper.conf: 2)
wrapper.conf: wrapper.app.parameter.4 to .7	Optional, in this order: number of connection attempts per RMI server, maximum total number of RMI servers (only with object pool), use object pool (true/false), start method of the RMI servers (ant, jexec or wrapper).	default: 20 attempts, 20 servers, object pool false, start method ant

Vacation: accounts, reference date and absence reasons

These keys control the display of remaining vacation and of planned and unplanned vacation. **Note:** If the remaining vacation account in days or the one in hours is empty, the values for planned/unplanned vacation are not calculated at all; a warning appears in the connector log ("Remaining Account Holidays not set"). Both accounts must therefore be set, even if calculation is only done in days. The page Berechnung und Darstellung von verplanten/unverplanten Urlaub[4] describes the calculation and display.

Property	Description	Possible value(s)
TaJournalDAO.remainingAccountHolidays	**Required:** yes. Conditional: required when remaining or planned/unplanned vacation is displayed. Name of the IF6020 account (vkkonto<no>) holding the remaining vacation in days. Mask: "Account for remaining holidays (d)". The shipped file sets vkkonto4.	account name (default: empty; shipped file: vkkonto4)
TaJournalDAO.remainingAccountHolidays2	**Required:** yes. Conditional as above. Name of the IF6020 account holding the remaining vacation in hours. Mask: "Account for remaining holidays (h)". The shipped file sets vkkonto3.	account name (default: empty; shipped file: vkkonto3)
TaJournalDAO.holidayEntitlement	**Required:** yes. Conditional as above. Name of the IF6020 person field (master record) that holds the vacation entitlement added every year. Mask: "Personfield for holiday entitlement". The shipped file sets stwert5.	field name (default: empty; shipped file: stwert5)
TaJournalDAO.holidayAbsenceReasons	**Required:** yes. Conditional as above. Comma-separated list of the absence reason numbers that count as vacation (for planned/unplanned vacation and the vacation calculations). Mask: "Vacation absence reasons". The former key	list, e.g. 1 (default: empty; shipped file: 1)

	TaJournalDAO.allowedAbsenceReasons is renamed to this key at startup.	
TaJournalDAO.entryDateForAccountHolidays	**Required:** no. If true, the entry date is the system-wide default reference date for the yearly addition of the vacation entitlement, unless something else is stored for the person. Mask: "Entrydate is vacation entitlement reference date".	true/false (default: false)
TaJournalDAO.alternativeDateForAccountHolidays	**Required:** no. Fixed system-wide vacation reference date as a fallback. Format DD.MM. or DD.MM.YYYY; without a year the current year is added. Mask: "Alternative Vacation Entitlement Reference Day".	date (default: empty, then 1 January)
TaJournalDAO.taPersonFieldnameAlternativeDateForAccountHolidays	**Required:** no. Name of the person field containing the vacation reference date of the individual person. Mask: "Personfieldname with vacation entitlement day".	field name (default: empty)
TaJournalDAO.taPersonFieldnameFlagEntryDateForAccountHolidays	**Required:** no. Name of a person field that must contain the value J for the entry date to be the reference date of this person. Mask: "Boolean person fieldname for entitlement addition on entrydate".	field name (default: empty)
TaJournalDAO.infoFieldTimeModelHalfDayVacation	**Required:** no. Name of a field in the daily program that marks with J daily programs for days such as 24 December or 31 December that count only as half a vacation day for a full-day vacation. Mask: "Marker-field in daily program for special public holidays which account for a half-day vacation". Empty: function off.	field name (default: empty)

Order in which the reference date for the yearly addition of the vacation entitlement is determined (the first matching rule applies):

1. The person field from taPersonFieldnameFlagEntryDateForAccountHolidays contains J: entry date of the person.
2. The person field from taPersonFieldnameAlternativeDateForAccountHolidays contains a date: that date.
3. entryDateForAccountHolidays is true: entry date of the person.
4. alternativeDateForAccountHolidays is set: that day and month.
5. Otherwise: 1 January.

Vacation entitlement and levels

These keys (mask group "Vacation Calculation Binding") define from which person fields and accounts the yearly vacation entitlement is determined. The entitlement depends on the years of service: level 1 applies as long as the years of service are below the value of level 2, then level 2, and from the value of level 3 on level 3. Levels 2 and 3 are only read if both the entitlement field and the field with the years of service of the level are set (level 3 additionally only if level 2 is set). The years of service are counted from the entry date.

Property	Description	Possible value(s)
TaVacationEntitlementDAO.taPersonFieldEntryDate	**Required:** optional (yes, if the vacation entitlement is calculated; if empty, the mask help says the field steindat is used). Name of the person field with the entry date used for years of service and vacation reference date. Mask: "Personfieldname for entrydate" (group "Person Settings"). If no date can be read, the start of validity of the person record is used.	field name (default according to the mask help: steindat)
TaVacationEntitlementDAO.taPersonFieldHolidayEntitlement	**Required:** optional (yes, if the vacation entitlement is calculated from person fields). Person field with the yearly vacation (days) of level 1. Mask: "Personfield for holiday entitlement (1st level)". If the field is empty or 0, 25 days apply.	field name (default: empty; fallback 25 days)
TaVacationEntitlementDAO.taPersonFieldHolidayEntitlementLevel2	**Required:** optional. Person field with the yearly vacation (days) of level 2. Mask: "Personfield for holiday entitlement (2nd level)". If the field is empty or 0, 30 days apply.	field name (default: empty; fallback 30 days)
TaVacationEntitlementDAO.taPersonFieldHolidayEntitlementLevel2Years	**Required:** optional. Person field with the number of years of service from which level 2 applies. Mask: "Personfield for No of employment years for Holidays Level 2". If the field is empty or 0, level 2 is never reached.	field name (default: empty)
TaVacationEntitlementDAO.taPersonFieldHolidayEntitlementLevel3	**Required:** optional. Person field with the yearly vacation (days) of level 3. Mask: "Personfield for holiday entitlement (3rd level)". If the field is empty or 0, 35 days apply.	field name (default: empty; fallback 35 days)
TaVacationEntitlementDAO.taPersonFieldHolidayEntitlementLevel3Years	**Required:** optional. Person field with the number of years of service from which level 3 applies. Mask: "Personfield for No of employment years for Holidays Level 3".	field name (default: empty)

TaVacationEntitlementDAO.taPersonFieldUnit	**Required:** no. Person field that defines the calculation unit of the vacation calculation. If the field has the value S, hours are used, with any other or empty value days. Mask: "Person field for holiday entitlement (calculation unit)".	field name (default: empty = calculation in days)
TaVacationEntitlementDAO.taHistoricHolidayEntitlement	**Required:** no. Name of the account or daily value in which the accounting stores the vacation entitlement (level 1) historically. For reference dates in the past the entitlement is read from there on the respective vacation reference date; if the value cannot be read, the person fields are used. Mask: "Account/Daily value for historic holiday entitlement".	account/value name (default: empty)
TaVacationEntitlementDAO.taHistoricHolidayEntitlementLevel2	**Required:** no. As above for level 2. Mask: "Account/Daily value for historic holiday entitlement level 2".	account/value name (default: empty)
TaVacationEntitlementDAO.taHistoricHolidayEntitlementLevel3	**Required:** no. As above for level 3. Mask: "Account/Daily value for historic holiday entitlement level 3".	account/value name (default: empty)
TaVacationEntitlementDAO.vacationRemunerationAccount	**Required:** no. Account (continuous yearly) for vacation remuneration, e.g. as compensation for negative balances or paid-out vacation days. Taken into account in the display of the remaining vacation development. Mask: "Account holiday remuneration". Empty: no remuneration.	account name (default: empty)
TaVacationEntitlementDAO.vacationRemunerationMonthlyAccount	**Required:** no. Monthly account for the correction of the vacation remuneration. Mask: "Account holiday remuneration correction (mthly)". Empty: no correction.	account name (default: empty)

Persons: synchronization, info fields and deletion

Webdesk creates a person master record in the IF6020 for each person (for each employment in case of multiple employments). These keys define in which IF6020 info fields name, employee number and person UID are stored. The info fields must exist in the IF6020, be unused and long enough (notes and check script on Breaking Changes[5]). Existing data is transferred to the configured fields with the job migrateEmployeeId[6].

Property	Description	Possible value(s)
TaPersonDAO.fnFirstName	**Required:** yes. IF6020 info field holding the first name (by default the IF6020 only knows a compound name). Mask: "Fieldname for firstname". The shipped file sets stpinfo120.	field name (default: empty; shipped file: stpinfo120)
TaPersonDAO.fnLastName	**Required:** yes. IF6020 info field holding the last name. Mask: "Fieldname for lastname". The shipped file sets stpinfo118.	field name (default: empty; shipped file: stpinfo118)
TaPersonDAO.nameOfFieldStoringPersonUid	**Required:** yes. Conditional: to be set up for connections to Webdesk 4.64 and later according to Breaking Changes[7]. Pstamm info field in which the person UID (PoPerson) is stored. Mask: "Name of field storing PoPerson UID" (read-only, change only in the file). The mask help names at least 20 characters, the page Breaking Changes a field length of 50 characters.	field name (default: empty)
TaPersonDAO.employeeIdField	**Required:** yes. Conditional as above. Pstamm info field holding the real Webdesk employee number (not the standard field stpersnr), at least 20 characters long. Mask: "Name of the field storing the employee ID" (read-only). The default is deliberately empty so that the field has to be configured.	field name (default: empty)
TaPersonDAO.effectiveEmploymentMarkerField	**Required:** yes. Conditional as above. Pstamm info field (1 character) with the flag J/N saying whether the person record is the currently effective employment. Used when searching by employee number. Mask: "Name of the field storing the effective employment marker" (read-only). The default is deliberately empty.	field name (default: empty)
TaPersonDAO.taIdOfTemplatePerson	**Required:** no. TA ID of a template master record that is copied when a new person is created if no other template is specified. Mask: "Template ta-ID for creations".	TA ID (default: empty)

TaPersonDAO.plausiPliNumberForSaveAndDelete	**Required:** no. Number of an IF6020 PLI script ("PlausiPli") that is run on every save (arguments: TA ID and insert or update) and before deleting a person (delete). The script can check mandatory fields and prevent saving. Mask: "ID of PlausiPli". Empty: no script.	PLI number (default: empty)
TaPersonDAO.disableOrgUnitCheckOnSave	**Required:** no. If true, saving does not check whether the person's department exists in the IF6020. Useful if persons are administered in Webdesk and the departments are not maintained in the IF6020. Mask: "Disable orgunit (department) check on save?".	true/false (default: false)
TaPersonDAOImpl.pliNumberToRunOnTerminationDateChange	**Required:** no. IF6020 script (PLI) that is run after the termination date of a person has actually changed; also when the termination date is cleared (then with an error log only if the script fails). Mask: "PLI-Number which will be run on termination date change.". Without a number nothing is run. Warning: the key starts with TaPersonDAOImpl, not TaPersonDAO.	PLI number (default: empty)
TaPersonDAOImpl.pliArgumentsToRunOnTerminationDateChange	**Required:** no. Arguments for the script, separated by ;. The placeholder $taId is replaced by the TA ID of the saved person. Mask: "PLI-Arguments which will passed to the script".	list, e.g. $taId;1 (default: empty)
TaPersonDAO.additionalTablesForTaPersonDeletion	**Required:** no. Additional IF6020 tables with person data that are also deleted when a master record is deleted from Webdesk. Per table: table name, field with the TA ID and optionally a date field, separated by ,; separate several tables with ;, e.g. tableA, ta_id_field, date_field; tableB, ta_id_field. Without a date field all rows of the person are deleted, with a date field only rows with a date after 1 January 2000. Mask: "Extra tables with personinfos".	text (default: empty)

Bookings and shift change

These keys concern bookings written from Webdesk to the IF6020 (mask group "Booking Parameters").

Property	Description	Possible value(s)
TaBookingDAO.terminal	**Required:** no. Number of the IF6020 terminal stored as reference for bookings from Webdesk. If the person has a time zone, the matching virtual terminal of that time zone is used instead; if there is none, the booking is rejected with an error. Mask: "ID of terminal for Webdesk bookings".	terminal number (default: 1000 if empty; shipped file: 1)
TaBookingDAO.allowDirectSwitchingToAbsenceCodesWithAttendantBehavior	**Required:** no. If true, an open presence or absence interval (only a "coming" booking) can be switched directly to another absence reason with attendant behavior. If false, an error message is shown. Mask: "Allow direct switching to absence reasons with attendant behavior".	true/false (bean default: true; if the key is missing, false applies at startup)
TaBookingDAO.logDeletionsToInvalidBookings	**Required:** no. If true, deletions and changes of bookings are logged in the invalid bookings of the IF6020 and could theoretically be restored there. Mask: "Log deletions in the 'invalid' bookings?".	true/false (bean default: true; if the key is missing, false applies at startup)
TaBookingDAO.enableTimeModelShift	**Required:** no. If true, a booking from Webdesk automatically selects the matching daily program (shift) and sets it for the day (prerequisites: valid shift change authorization of the person and shift windows for coming/going in the daily programs). Mask: "Automatic shift-change based on booking". **Restart required:** the change is written to the file on saving, but applied in the running connector only after a restart.	true/false (default: false)
TaJournalDAO.keepBookingMessagesInCorrection	**Required:** no. If true, booking messages are displayed in the journal and the correction client as correction comments. Mask: "Booking messages are returned as correction comments".	true/false (default: false)
TaBookingDAO.ipToTerminalAsString	**Required:** no. Not in the mask. Assigns terminals to IP ranges:	text (default: empty)

	entries of the form <terminal>:<IP prefix>, separated by ;, e.g. 2:192.168.0;3:10.. The best matching IP prefix applies; if none matches, the terminal from TaBookingDAO.terminal applies. Only effective for persons without a time zone. Read when the connector starts.	

Working time accounts and maximum working time

These keys (mask group "Accounting Parameters") control the display of the current working time and the check against the daily and weekly maximum working time (see job checkMaxWorkingTime[8]). The current working time is determined by placing the bookings of the day so far on a dummy person record and accounting them there.

Property	Description	Possible value(s)
TaBrokerImpl.personTaIdsReservedForCalculations	**Required:** yes. Conditional: required for the functions current daily and weekly working time; without an entry these functions report a configuration error. Comma-separated list of TA IDs of dedicated person master records used solely for these calculations (several allow parallel calculations). The records must not be used for anything else; they are excluded from the data warehouse export. Mask: "Person Dummy-TA-ID(s) for working-time calculations". The former key TaAccountDAO.taIdOfDummyRecordForCalculations is renamed at startup.	list of TA IDs (default: empty)
TaAccountDAO.infoFieldsToCopyForCalculations	**Required:** no. Person info fields relevant for the calculation, which are therefore copied to the dummy record before the calculation. Mask: "Person info-fields for Dummy".	list of field names (bean default: stwert1, stwert2; if the key is missing, an empty list applies at startup)
TaAccountDAO.workingTimeAccount	**Required:** no. Account holding the accounted daily working time, also used for the check of the daily maximum working time. Mask: "Account for daily working time".	account name (default: twistzeit, also if empty)
TaAccountDAO.weeklyWorkingTimeAccount	**Required:** no. Account holding the accounted weekly working time, also used for the check of the weekly maximum working time. Mask: "Account for weekly working time".	account name (default: vkwoche, also if empty)

TaAccountDAO.infoFieldTimeModelMaxDailyWorkingTime	**Required:** no. Daily program with the maximum daily working time (industrial hours/ minutes). Takes precedence over the person field. Mask: "Daily Programm field for max daily working time".	field name (default: empty)
TaAccountDAO.infoFieldMaxDailyWorkingTime	**Required:** no. Person field with the maximum daily working time; fallback if the daily program provides no value. If no value is stored anywhere, 10 hours apply. Mask: "Personfield for max daily working time".	field name (default: empty; fallback 10 hours)
TaAccountDAO.infoFieldTimeModelMaxWeeklyWorkingTime	**Required:** no. Weekly program with the maximum weekly working time. Takes precedence over the person field. Mask: "Weekly Programm field for max weekly working time".	field name (default: empty)
TaAccountDAO.infoFieldMaxWeeklyWorkingTime	**Required:** no. Person field with the maximum weekly working time; fallback if the weekly program provides no value. If no value is stored anywhere, 50 hours apply. Mask: "Personfield for max weekly working time".	field name (default: empty; fallback 50 hours)
TaAccountDAO.doubleReaccountingRequired	**Required:** no. If true, the day is re-accounted twice on an account correction. Only for installations with custom accounting scripts that deliver the correct result only after double accounting; costs additional processing time. Mask: "Use double reaccounting on account correction" (appears in the groups "Holiday Display (planned/ unplanned)" and "Accounting Parameters").	true/false (default: false)

Absence reason lists and calendar

Note on the lists of absence reason numbers in this section and in the vacation section: the absence reason numbers defined in the IF6020 are used. In the mask they are selected from the existing absence reasons.

Property	Description	Possible value(s)
TaAbsenceReasonSettings.idsOfProductiveAbsenceReasons	**Required:** no. Comma-separated IDs of the absence reasons that count as productive (count as work). Maintained by Webdesk: entered at Webdesk startup if empty, and updated when an	list of IDs (default: empty; filled automatically)

	absence reason is saved in the action editAbsenceReason. Mask: "IDs of productive absence reasons" (read-only). Do not change by hand.	
TaAbsenceReasonSettings.idsOfAbsenceReasonsWithAttendantBehavior	**Required:** no. Comma-separated IDs of the absence reasons with attendant behavior (e.g. remote or home work). Maintained by Webdesk like the previous list. Mask: "IDs of absence reasons with attendant behavior" (read-only). The behavior is defined on the absence reason itself (see Systemparameter TA[9]).	list of IDs (default: empty; filled automatically)
TaCalendarDAO.absenceCodesForSickLeave	**Required:** no. Absence reason numbers for sick leave. Basis for the sum of sick days per month in the calendar. Mask: "Absencereasons for Sick-Leave".	list (default: empty)
TaCalendarDAO.absenceCodesForSpecialLeave	**Required:** no. Absence reason numbers for special leave (e.g. moving, death in the family). Basis for the sum of special leave days per month in the calendar. Mask: "Absencereasons for Special Leaves".	list (default: empty)
TaCalendarDAO.repairCalendarIntervalsOnTheFly	**Required:** no. If true, absence intervals from the IF6020 are compared with the original requests (process IDs) before being returned and split again if necessary. Background: when entering absences, adjacent intervals with the same absence reason may be merged automatically. Mask: "Keep absence intervals in the calendar consistent to Webdesk requests".	true/false (default: false)
TaAbsenceIntervalDAO.taCalendarSuffixFuerUntertgFehlzeiten	**Required:** no. Shortname of the custom IF6020 table for passive, intra-day absences (e.g. f1). Requires an accounting extension on the Interflex side that writes to this table. Mask: "Shortname custom table for underday absences". Empty: function off. The former key TaCalendarDAO.taCalendarSuffixFuerUntertgFehlzeiten is renamed at startup.	short name (default: empty)

Attendance board

These keys (mask group "StatusList Configuration") determine how the status of persons is determined in the attendance board. The lists contain numbers of daily or weekly programs, separated by commas.

Property	Description	Possible value(s)
TaStatusTableauDAO.tpnrNotBooking	**Required:** no. Numbers of the daily programs for persons who do not book ("non-bookers"). A person without a booking who has one of these daily programs gets the status "Person is not booking" instead of "absent". Mask: "List of daily programs used by Non-Bookers". The shipped file sets -1 (no program).	list (default: empty; shipped file: -1)
TaStatusTableauDAO.tpnrAbsenceOnlyBooking	**Required:** no. Numbers of the daily programs for absence-only recorders (negative time recorders). Their absences are displayed as for normal time recorders; without an absence the status is "Person is not booking". Mask: "List of daily programs used by Absence-only TA Management".	list (default: empty)
TaStatusTableauDAO.wpnrAbsenceOnlyBooking	**Required:** no. As above, but numbers of weekly programs. Mask: "List of weekly programs used by Absence-only TA Management".	list (default: empty)
TaStatusTableauDAO.statusPresentForAbsenceCodesWithAttendantBehavior	**Required:** no. If set, persons are also shown as present when they are currently booked with an absence reason with attendant behavior. Mask: "Status present for absences with attendant behavior".	true/false (default: false)

Note: Entries that you delete from the three program lists in the mask are not removed in the running connector (new entries are only added). The deletion takes effect only after a restart of the service.

Number ranges for daily and weekly programs

When Webdesk creates new daily or weekly programs in the IF6020, the first free number starting from the value given here is assigned (mask group "Timemodel Administration").

Property	Description	Possible value(s)
TaTimeModelDAO.minId	**Required:** no. Lower threshold for the numbers of newly created daily programs. Mask: "Minimal ID for day programs". Empty: next free number from 1.	integer (default: 1)

TaTimeModelProgramDAO.minId	**Required:** no. Lower threshold for the numbers of newly created weekly programs. Mask: "Minimal ID for week programs". Empty: next free number from 1.	integer (default: 1)

Data warehouse

Optionally, the time and attendance module can read part of the time data from a data warehouse (SQL database) instead of from the IF6020. The export is configured through the file synchronization-job.xml in the conf directory of the connector; if it is missing, nothing is exported. The ongoing synchronization is done by the job taIf2sqlSynchronizationJob[10].

Property	Description	Possible value(s)
TaDataWarehouseDAO.warehouseEnabled	**Required:** no. Switches the use of the data warehouse on. Mask: "TA Data Warehouse Enabled". Switching takes effect immediately in the broker. If a database error is detected while building the warehouse, the connector switches the use off by itself.	true/false (default: false)

Further keys, file only

Besides the keys described above, the connector evaluates every entry of the form beanname.property for which a bean and a settable property exist (see the section on how the parts work together). In addition, the following entry of the shipped file is of interest:

Property	Description	Possible value(s)
TaCalendarDAO.dayProgsThatAvoidAbsenceMerge	Comma-separated daily program numbers that prevent consecutive absences from being merged. Only concerns the old absence store and is marked as deprecated in the code. The shipped file sets 1,2.	list (default: empty)

Automatic adjustments at startup

At startup the broker renames or removes the following obsolete entries and writes the file back:

Old key	Action
TaJournalDAO.allowedAbsenceReasons	becomes TaJournalDAO.holidayAbsenceReasons
TaAccountDAO.taIdOfDummyRecordForCalculations	becomes TaBrokerImpl.personTaIdsReservedForCalculations
TaCalendarDAO.taCalendarSuffixFuerUntertgFehlzeiten	becomes TaAbsenceIntervalDAO.taCalendarSuffixFuerUntertgFehlzeiten
TaPersonDAO.bookingMessageField	is removed

When the connector is updated with the installation package, the installer carries the entries of the old tacon.properties over into the new file; keys of the same name keep their previous value, new keys of the new file are kept. Back up the file before every update (see Update mit Installpackage + Patches[11]).

Setup and verification

1. Stop the service "Webdesk TA Connector for IF6020" and back up tacon.properties.
2. Enter at least: TaRmiService.username and TaRmiService.password (technical IF6020 user), the accounts and fields of the vacation calculation, TaPersonDAO.fnFirstName and TaPersonDAO.fnLastName, with Webdesk 4.64 and later the three info fields for person UID, employee number and effective-employment marker, and if needed TaBrokerImpl.personTaIdsReservedForCalculations.
3. Start the service. The log wrapper.log must show the message "TA RMI Broker successfully started with N running servers of M requested".
4. Open the mask "Systemkonfiguration" in Webdesk, check the remaining values and save once. This writes all keys of the mask to the file.
5. The credentials are encrypted at the next start of Webdesk (see above); afterwards check in the mask that "Credentials encrypted" is active.

Note: Typos in a key are not reported; the entry simply has no effect. When behavior is unexpected, first check the key name against this page. Errors when starting the connection are in the connector log (wrapper.log in the logs directory).

1. /daisy/webdesk-manual-admin/7507-dsy.html?language=4
2. /daisy/webdesk-manual-admin/5120-dsy.html?language=4
3. /daisy/webdesk-manual-admin/10823-dsy.html?language=4
4. /daisy/webdesk-manual-admin/3612-dsy.html?language=4
5. /daisy/webdesk-manual-admin/11499-dsy.html?language=4
6. /daisy/webdesk-manual-admin/11536-dsy.html?language=4
7. /daisy/webdesk-manual-admin/11499-dsy.html?language=4
8. /daisy/webdesk-manual-admin/6281-dsy.html?language=4
9. /daisy/webdesk-manual-admin/5120-dsy.html?language=4
10. /daisy/webdesk-manual-admin/11538-dsy.html?language=4
11. /daisy/webdesk-manual-admin/6343-dsy.html?language=4