Zonabit Sistemi Srl – February 2015

How to plan Staff Shifts using

ZonaTEAM

SERVER VERSION

This document is distributed free. It gives directions to use the Software "ZonaTEAM Server Version" as it is currently available on http://www.zonabit.it. No warranty whatsoever is granted the users of this guide, of ZonaTEAM Server Version and of the source code of ZonaTEAM Server Test Client Application.

Index

ZonaTEAM Desktop and Server Version....4

ZonaTEAM Desktop and Server Version....4

ZonaTEAM Server....5

ZonaTEAM Server....5

ZonaTEAM Server....5

Test Client Application....5

Test Client Application....5

Test client application sources....5

Server Protocol Specifications....6

Server Protocol Specifications....6

ZonaTEAMServer installation and start....6

Data Encryption and Decryption....6

Workflow....7

Calls to Server - general format....7

Calls to Server - messages....8

Calls to Server - file transfer to Server flow....11

Configuration File Specifications....13

Configuration File Specifications....13

Configuration file format....13

General tags....13

Tag: !rem....13

Tag: !unique....13

Tag: !seqperx....13

Tag: !periodo....16

Tag: !feste Tag:!semifeste....16

Tag: !tipoturno....17

Tag: !squadra....17

Tag: !reparti....18

Tag: !qualifiche....19

Tag: !depositi....19

Tag: !coprire....20

Employment Contracts tags....22

Tag: !contratto....22

Tag: !copausapranzo....23

Tag: !costraordinario....24

Tag: !conotturno....25

Tag: !codovuto....26

Tag: !cowke....26

Tag: !coorvc....27

Tag: !coorpa....28

Tag: !coorpb....28
Tag: !comwl....28
Employees tags....29
Tag: !u....29
Tag: !uco....29
Tag: !ustoria....30
Tag: !ucondizioni....30
Tag: !uprefo....31
Tag: !uprefg....32
Tag: !upremored....33
Tag: !uprero....33
....34
Tag: !upretuo....34
Tag: !ulimtutor....35
Tag: !ulimorvie....35
Tag: !uviaggia....36
Tag: !uprevarie....37
Tag: !uimpo....37
Tag: !uferpe....38
Tag: !uperxe....39
Tag: !urq....40
PLANNING DIFFERENT PERIODS OF SCHEDULING 41
PLANNING DIFFERENT PERIODS OF SCHEDULING 41
DATA TO COLLECT BEFORE CREATING A CONFIGURATION....42
DATA TO COLLECT BEFORE CREATING A CONFIGURATION....42

ZonaTEAM Desktop and Server Version

ZonaTEAM is a powerful processing engine to schedule the shifts of any kind of workforce. ZonaTEAM uses an original heuristic set of algorithms to process a set of requirements of a staff to be employed under given conditions, and gives a highly optimized result.

ZonaTEAM Desktop is a desktop applications to schedule the shifts of any kind of workforce. ZonaTEAM Desktop has an intuitive user interface fit for small organizations and for individual productivity of a person having the task of scheduling the workforce activities.

But ZonaTEAM processing engine can be also used as a component of more complex and multi user software packages to schedule the activities of medium and big organizations.

ZonaTEAM Server is a Server application that can be used to schedule a staff through the same processing engine used by ZonaTEAM Desktop. ZonaTEAM Server has no user interface for the scheduling configuration data: it can be accessed via TCP-IP sockets and can be easily integrated in any ERP (Enterprise Resource Planning) system.

ZonaTEAM Desktop and ZonaTEAM Server can both be downloaded for free from http://www.zonabit.it . Both the systems can be used freely to plan a small test configuration. The usage with a bigger configuration requires the purchase of a monthly or annual license.

This documents contain the information needed to exchange data between an end user application (a component of an ERP system) and ZonaTEAM Server.

Detail:	ZonaTEAM Desktop and ZonaTEAM Server are currently available for Microsoft Windows (R). In the future a Linux version will be released. ZonaTEAM Server can be installed on a Windows Machine of your LAN or accessible through the internet, so the ERP application that interfaces it can be run on any other OS.
Focus:	In order to understand gradually ZonaTEAM concepts, you are advised to proceed in this way: • Download ZonaTEAM Desktop application, install it and watch the available test configuration which you will find installed together with the program. • Read the tutorial: How to plan Staff Shifts using ZonaTEAM, which is available from the first page of ZonaTEAM Desktop. This tutorial is for the Desktop version users, but if will be useful to understand the meaning of configuration data that you will send to ZonaTEAM Server. • Download ZonaTEAM Server and install it. • Download the sources of the Test Application and compile them on a Windows machine or on any other OS. The Test Application is delivered as a Windows Visual Studio Project, but the C++ core modules can be compiled with any C++ compiler. The Test Application has an essential GUI that can be substituted by any other user interface. • Access ZonaTEAM Server via your compiled version of the test application, process the test configuration available inside the source code and get the result. When the above steps will have been completed successfully, you will be ready to interface ZonaTEAM Server from your ERP system: build your configuration to be scheduled and process it via ZonaTEAM Server.

To resolve any doubt: www.zonabit.it assistenza@zonabit.it

ZonaTEAM Server

	ZonaTEAM Server
	ZonaTEAM Server
ZonaTEAM Server and ZonaTEAM Desktop are both available for free download from www.zonabit.it. Both applications can be used free to manage a small staff; for a bigger staff you need a license. ZonaTEAM Server has no user interface: the main page of the application allows you to choose the port for communication, and to start the protocol. When you purchase a license code, you must initialize a new license, or renew an existing one, through the user interface of ZonaTEAM Server. This are the only operation to do on ZonaTEAM Server. You can start multiple instances of ZonaTEAM Server on a machine. Instances after the first one will detect automatically their port for communication. Every instance of your client application will hook an instance of ZonaTEAM Server in order to process simultaneously different configurations, or to process simultaneously the same configuration: in this case you will get more solutions of the same problem, and you will the choose the best optimized one.	
Tip:	Also if you are interested only in ZonaTEAM Server, download and install both the Server and Desktop version. ZonaTEAM Desktop will allow you to view the data uploaded to ZonaTEAM Server by your client application, and this will be a powerful debugging tool.
Detail:	

Test Client Application

	Test program
	Test client application sources
The sources of a simple client program are available. The sources contain methods to send messages to ZonaTEAM Server, and can be used in the implementation of your client application.	
Focus:	This guide contains references to examples inside the test program, so be sure to have it available while reading.
Detail:	The test program contains the following modules: READ_ME.cpp = an empty source file, containing just comments. About.cpp, stdafx.cpp, ZTuWinServerTest.cpp = standard Windows MFC application modules (not to be used outside Windows Visual Studio Projects). About.h, resource.h, ZTuWinServerTest.h = standard Windows MFC header modules (not to be used outside Windows Visual Studio Projects). ZTuWinServerTestDlg.cpp, ZTuWinSERVERTestDlg.h = a Dialog calling all the available methods. Refer to the events linked to every button to write your application. ZtuWinServerPublic.h = definitions shared with ZonaTEAMServer. Include this module in your project and never change it. ZTuWinServerLIB.cpp, ZtuWinServerLIB.h = available methods to interface the Server. Compile this module in your environment, as a part of your test and final application. ZtuWinServerCypher.cpp = a simple data encryption technique shared with ZonaTEAMServer. Include this module in your project and never change it.

	Example_1.cpp = some correct Scheduling Configurations examples. They can be processed by ZonaTEAMServer.
Error to avoid:	

Server Protocol Specifications

	Server Protocol Specifications
	ZonaTEAMServer installation and start
ZonaTEAMServer can be installed on any recent Windows version.	
Example:	
Focus:	
Detail:	Just download the setup and install ZonaTEAMServer. The program should have full access to the folder: C:\ZonaTEAMdati (under root of the default disk). So be sure to grant the user full access to this folder. Ask your system administrator in case of doubt. The machine will be accept incoming calls on the LAN or internet, so configure the firewall in order to allow incoming calls. If you process a real configuration, you will need a license and an internet connection to validate it. If you process just a test configuration, the internet connection is not mandatory. The Server will be accessed via your LAN. Choose a port for the incoming calls and configure it on ZonaTEAM Server. We suggest port 9200 as base of your port array. Open Server with the corresponding button. The IP address of the Server on your LAN and on the internet will be shown. The Server will ping itself automatically at the LAN IP address every minute to check the network. You can start multiple instances of ZonaTEAMServer. Every instance will use two ports. Configure the base port when starting the first instance; further instances will recognize automatically their ports.
Error to avoid:	The test and final application can address ZonaTEAMServer via its address on the LAN or on the internet. If you choose to work via the internet, your router should be configured in order to link calls to the chosen port to the IP of ZonaTEAMServer on the LAN. Example: You chose base port = 9208 and opened the Server. The Router IP is 87.20.111.37 and the LAN IP is 192.168.1.3 (this is shown on ZonaTEAMServer after opening communication). The engaged port are 9208 and 9209. If you choose to address ZonaTEAMServer via the Router IP address 87.20.111.37, you should configure the router specifying that calls to port 9208 and 9209 must be addressed to the LAN IP address 192.168.1.3.

	Server Protocol Specifications
	Data Encryption and Decryption
Messages exchanged between ZonaTEAM Server and the client application are encrypted and decrypted via the methods available in the module ZtuWinServerCypher.cpp of the test program.	

Example:	
Focus:	If you want to use a more complex and personalized data encryption algorithm, address to us to implement it. A personalized release of ZonaTEAM Server will be delivered to you.
Detail:	The messages sent to ZonaTEAM must be put in an encoded frame through a call to: YSS_message_encode_binary (char *message, char *encoded, int len). The messages received from ZonaTEAM must be decoded through a call to: YSS_message_decode (char *message, char *decoded, int maxlen).
Error to avoid:	

	Server Protocol Specifications
	Workflow
Your application sends to ZonaTEAM Server some simple messages in order to open a communication, send a configuration file and a license number, process the file and get the processed result.	
Example:	
Focus:	You can start multiple instances of ZonaTEAM Server, and multiple instances of the Client application can look for a free Server instance for their use.
Detail:	The workflow is: The Client pings an instance of the Server in order to check if the Server answers on the network and is available. The Client reserves the Server for its own use, so that other instances of the Client cannot use the reserved instance of the Server. The Client sends a configuration file describing a set of scheduling requirements to be processed. The format of a valid configuration file is described in a further section of this document. The Client checks the format of the configuration file: the Server answers whether the file format is correct or not. The Client validates its user's license. If you have no valid license, the Server will process only some persons in the configuration file, for test purposes. The Client gives an order to process the sent configuration for a given time. While processing, the Client iterates the request of the status of processing. The Client waits for the processing end, or stops the processing. The Client gets the processed data: this means that the Server will send to Client the same data of the previous configuration file, enriched with data of the engagements of the staff. Processing can be repeated for a longer time in order to get a more optimized result.
Detail:	The Server disconnects the client after a minute of silence. So if you send a message more than a minute later than the previous one, or if you get no answer, you must ping again the Server. Before beginning a sequence of message it is good practice to ping the Server anyway.

	Server Protocol Specifications
	Calls to Server - general format
Example:	
Focus:	
Detail:	Every message sent to the Server has the format: CiiiiiiiMessageData , where:

	C = type of the Message, 1 byte iiiiiiii = identifier of the client, 8 bytes. MessageData = binary message data, specific for every message
Error to avoid:	

	Server Protocol Specifications
	Calls to Server - messages
Refer to the module ZTuWinServerLIB.cpp of the test client program sources to see examples of calls to Server. In every message you send a request to Server and wait for an answer, with a time out. Generally wait the answer for 5 seconds, which is a very large time out value, given the small size of messages. Wait for a longer time if you access the Server through the internet and communication is poor.	
Example:	

Focus:	`On generic errors, the Server answer is: YSS_NACK 'Y'`
	`// message not acknowledged`
	`If you get YSS_NACK, check carefully your code and address assistance if you don't understand the reason.`

Message:	Client message: YSS_PING 'A' // client pings Server (and gets identity) DATA: ‘A’ iiiiiiii Client_description The Client_description field is a descriptive string (optional, max. 40 bytes) that is displayed on the Server monitor and has no further use. NOTE: the first Ping Message must have iiiiiiii = “00000000”. The Server answer assigns to the client its identity to be used in every further message. Server answer: YSS_PONG 'a' DATA: ‘a’ iiiiiiii The answer to the first Ping returns iiiiiiii not equal to “00000000”. The Client application must store the returned iiiiiiii field and use it in every further call to Server.

Message:	Client message: YSS_RESERVE 'B'
	// client asks to reserve Server for its own use
	DATA: ‘B’ iiiiiiii Flag
	Flag values:
	‘1’ = try to reserve the Server, give up if already reserved by another instance of the Client
	‘2’ = force reservation
	Server answer: YSS_RESERVE_ANSWER 'b'
	// answer to reservation request
	DATA: ‘b’ iiiiiiii Status
	Status values:
	‘0’ = Server reserved OK
	Other values = Server could not be reserved

Message:	Client message: YSS_FILESESSION 'C' // client sends to Server a fiel containing the configuration to be processed

	(may be zipped) DATA: ‘C’ iiiiiiii Flag Flag is for further espansions (set to ‘1’) Server answer: YSS_FILEACK 'c' // ack to a file message DATA: ‘c’ iiiiiiii NOTE: for the subsequent file transfer messages see next section.

Message:	Client message: YSS_FILE_CHECK 'F'
	// client asks to check the file sent in filesession
	DATA: ‘F’ iiiiiiii Name_on_server
	Name_on_server: the file name on Server side. Use a simple constant, e.g. “myconfiguration.txt”
	Server answer: YSS_FILE_CHECK_ANSWER 'f'
	// answer to file check request
	DATA: ‘f’ iiiiiiii Status Descriptive_text
	Status values:
	‘0’ = file checked, can be processed
	Other values = file not correct
	Descriptive_text = a descriptive text of the error inside the configuration file

Message:	Client message: YSS_USER_LICENSE 'G'
	// client sends to Server the user license to process the configuration
	DATA: ‘G’ iiiiiiii License_number-Lassword
	License_number= number of your license
	- = character ‘-‘
	Password = your license password
	Server answer: YSS_USER_LICENSE_ANSWER 'g'
	// Server tells if license is valid
	DATA: ‘g’ iiiiiiii Status Descriptive_text
	Status values:
	‘0’ = license checked, can be used for processing
	Other values = license not correct
	Descriptive_text = a descriptive text with information about the license (maximum staff and expiry, DD-MM-YYYY)
	Note: if the license is not valid, subsequent processing will use only 5 members of the staff and will give a partial result.

Message:	Client message: YSS_START_PROCESSING 'H'
	// client sends the order to process the previously sent configuration
	DATA: ‘H’ iiiiiiii Minutes
	Minutes = minutes of processing, a value between “1” and “1440” (1440 minutes = 24 hours)
	Server answer: YSS_PROC_STATUS_ANSWER 'h'
	// answer to processing status request
	DATA: ‘h’ iiiiiiii Status Descriptive_text

	Status values: '0' = processing started and currently running '1' = processing not running Descriptive_text = a descriptive text with detailed information about the current processing status NOTE: processing will last some time more (about 10% - 20% more) than the duration specified in your message, because when the processing scheduled time elapses ZonaTEAM will perform another entire cycle of optimization.

Message:	Client message: YSS_PROC_STATUS 'I'
	// client asks the current status of processing. Stop processing is an optional parameter
	DATA: | 'I' | iiiiiiii | Flag
	Flag values:
	'0' = just get processing status
	'1' = stop processing
	Note: if you send a stop processing order, processing will terminate after some time (up to 1 minute), because before terminating a short complete optimization cycle will be performed.
	Server answer: YSS_PROC_STATUS_ANSWER 'h'
	// answer to processing status request
	For description of this answer message see above.

Message:	Client message: YSS_GET_PROCESSED_DATA 'J'
	DATA: | 'J' | iiiiiiii | Flag1Flag2
	Flag1 values:
	'0' = get the error list (description of residual errors)
	'1' = get the processed data
	Flag2 values:
	'1' = get the first data packet
	'0' = get the next data packet
	Before sending this message, be sure that processing is not running (send YSS_PROC_STATUS).
	First get the first packet, then the subsequent packets, and store them in a file. The file will contain the same configuration information you sent to Server, plus the shifts assigned to staff members.
	// client asks the processed data, with parameter 'first' or 'next'
	Server answer: YSS_FILEPACK 'E'
	// file packets
	DATA: | 'E' | iiiiiiii | Packet _number|Flag|Packet_data
	Packet_number = packet number in a hexadecimal string (8 bytes). Accept the number of the first packet, than check the number of the subsequent packet: the number must be equal to the previous number + 1.
	| = character '|'
	Flag values:
	'0' = current packet is not the last one, ask for the next one
	'1' = current packet is the last one, do not ask for more

	| = character '|' Packet_data = packet data to be stored in a file (open as a binary file)

	Server Protocol Specifications
	Calls to Server - file transfer to Server flow

In every message you send a request to Server and just wait for an answer, with a time out.
When you upload a configuration file on the Server, the workflow is the following:
`Client sends YSS_FILESESSION`
`Server sends YSS_FILEACK (or Client gives up transmission)`
`Client sends YSS_FILEPROTO`
`Server sends YSS_FILEACK (or Client gives up transmission)`
Client divides the file into packets of `MIN_PACK_DATA size`
`Client sends YSS_FILEPACK`
`Server sends YSS_FILEACK (or Client gives up transmission)`
`...`
`until the last data packet has been sent.`	

Example:	
Focus:	

Message:	Client message: YSS_FILESESSION 'C'
	// client sends to Server a fiel containing the configuration to be processed (may be zipped)
	DATA: | 'C' | iiiiiiii | Flag
	Flag is for further espansions (set to '1')
	Server answer: YSS_FILEACK 'c'
	// ack to a file message
	DATA: | 'c' | iiiiiiii |

Message:	Client message: YSS_FILEPROTO 'D'
	// client sends to Server a fiel containing the configuration to be processed (may be zipped)
	DATA: | 'C' | iiiiiiii | Packet _number|File_name|File_id
	Packet_number = packet number in a Hexadecimal string (8 bytes). Set it to a random value. You can begin with 1 the first time, then increment it, then never reset it.
	| = character '|'
	File_name = file name on Server side. Use a mnemonic name, e.g. "Mydata.txt". Do not use the reserved values "Turni2.txt" and "Turni1.txt" and "TurniA.txt"
	| = character '|'
	File_id = for further use. Set to "ABCD".
	Server answer: YSS_FILEACK 'c'
	// ack to a file message

Message:	Client message: YSS_FILEPACK 'E' // file packets DATA: | 'E' | iiiiiiii | Packet _number|Flag|Packet_data Packet_number = packet number in a hexadecimal string (8 bytes). Increment the value by 1 at every new packet. If the value is not incremented, you will get YSS_NACK as answer.

	| = character '|' Flag values: '0' = current packet is not the last one, Server must wait for the next one '1' = current packet is the last one, Server must close the file | = character '|' Packet_data = packet data read from your configuration file (open your file as a binary file, avoid automatic conversions of characters) Server answer: YSS_FILEACK 'c' // ack to a file message

Configuration File Specifications

	Configuration File Specifications
	Configuration file format
The file transmitted to ZonaTEAMServer is a pure text file, where every line has the fomat: '!'tag'|'data ... data... '|' and is terminated by \n. Data inside a line can be divided by the characters: '|' or '-' or ':' , depending on the tag. So the user's names of objects should never contain the characters: '|' or '-' or ':'	
Example:	!reparti|Ambulatory-0--|
Focus:	
Detail:	
Error to avoid:	

General tags

	Configuration File Specifications **MANDATORY**
	Tag: !rem
!rem|*name*|	
Example:	!rem|Name of my Team configuration|
Focus:	
Detail:	**name: (40 characters string)** The first line of every configuration file must contain a !rem! tag followed by the name of the configuration. Other !rem| tags in the file are ignored
Error to avoid:	

	Configuration File Specifications **MANDATORY**
	Tag: !unique
!unique|*name*|	
Example:	!unique|Bristol_general_hospital_department_BCDEF|
Focus:	
Detail:	**name: (40 characters string)** Every file must contain a 40 characters unique identifier. Used to validate the license. If you have two or more different configurations give everyone an identifier and don't change it any more.
Error to avoid:	

	Configuration File Specifications **OPTIONAL**
	Tag: !seqperx
!seqperx|*field1*|*field2*|*field3*|*field4*|*field5*|*field6*|*field7*|*field8*|*field9*|*field10*|	
Example:	!seqperx|A0|B0|D0|C0|E0|F0|G0|H0|I0|J0|

Focus:	Optional parameter. Necessary if you want to change the processing sequence of phases. The processing of the planning first assigns shifts to persons without violating any expressed condition of the job contracts and any constraint imposed by planned holidays and absences, by personal preferences and by availability and mandatory presence. After this, the processing calculates the best possible result in phases (the phases which do not have configured requests are ignored):

Staff requirements satisfaction	Tries to satisfy 100% of the requests for coverage of shifts that have been configured.
Reduction of overtime and substitution **A**	Tries reduction to zero of overtime and tasks in substitution (wh an employee works with a lower qualification than his profession qualification).
Reduce to minimum cost by goal **B**	Control of Cost by Goal is used only in the case that some memb of the staff (such as managers or apprentices) have a fixed conventional monthly cost, while others have cost in proportion t the hours of work. If at least one staff member is configured wit conventional fixed monthly cost, the automatic processing of scheduling plays an optimization phase of Cost by Goal (Reductio to Minimum Cost by Goal), which reduces as much as possible th commitment of the staff whose cost is proportional to the worke time. If all the staff has cost proportional to the work done (default), t control of the Cost by Goal is not active.
Equable distribution of WORK + HOLYDAY Hours **C**	Tries to ensure that all are delivering their due working hours an enjoy the holidays they are entitled, without difference among persons.
Equable distribution of INTRA MOENIA Hours **D**	Seeks the equable distribution of hours Intramoenia (if controlle in relation to the declared preference for each person.
Equable distribution of free Weekends and Holydays and Bank Holydays **E**	Tries to distribute equably the free Weekends (Weekend limits a defined in contracts) and the free Holydays and Bank Holydays (defined as such in the Environment Configuration).
Equable distribution of dislikeable Shifts **F**	Seeks the fair distribution of shifts for which this is necessary (fo example, those at night), as declared in the configuration of the types of shifts.
Preferred distribution of Shifts **G**	Tries to meet the organization of consecutive shifts that may hav been configured in the preferences for each person.

	Minimization of vacant times among short Shifts **H**	The vacant times among shotr shifts can be configured in the Contracts. If configured, the vacant times are minimized in this phase.
	Ordered distribution of Shifts **I**	Tries to make uniform the time commitment of the persons, by assigning (for example) 3 mornings and then 3 afternoons, rathe than a morning, an afternoon, a morning, and so on.
	Composition of Teams **J**	Tries to form teams, if configured.
Detail:	This string represents the user's personalized sequence of phases in scheduling processing using the letter of each phase. **field1 : (letterinteger)** letter: name of the phase integer: unused set to 0 **field2 : (letterinteger)** letter: name of the phase integer: unused set to 0 **field3: (letterinteger)** letter: name of the phase integer: unused set to 0 **field4 : (letterinteger)** letter: name of the phase integer: unused set to 0 **field5 : (letterinteger)** letter: name of the phase integer: unused set to 0 **field6 : (letterinteger)** letter: name of the phase integer: unused set to 0 **field7 : (letterinteger)** letter: name of the phase integer: unused set to 0 **field8 : (letterinteger)** letter: name of the phase integer: unused set to 0 **field9 : (letterinteger)** letter: name of the phase integer: unused set to 0 **field10 : (letterinteger)** letter: name of the phase integer: unused set to 0	

Error to avoid:	

	Configuration File Specifications **MANDATORY**
	Tag: !periodo
!periodo|*date1*|*date2*|*date3*|	
Example:	!periodo|2015-9-30|2015-10-6|2015-1-1|
Focus:	Here there's start and end date of the next period you want to plan automatically (Automated Planning Period). If there are Shifts straddling the midnight, the end date is extended so that they can be included in Planning. For example, if the Planning goes from first to 30th April, and there are Shifts that last from 22:00 to 6:00, Shifts that begin at 22:00 on April 30 and end at 6:00 on May 1 are scheduled.
Detail:	**date1: (AAAA-MM-GG)** Beginning of the period to be planned automatically **date2: (AAAA-MM-GG)** End of the period to be planned automatically **date3: (AAAA-MM-GG)** Start date of the current year
Error to avoid:	

	Configuration File Specifications **OPTIONAL**
	Tag: !feste Tag:!semifeste
!feste|*year[|month-day|month-day|month-day|]* !semifeste|*year[|month-day|month-day|month-day|]*	
Example:	!feste|2015|4-6|4-25|5-1|6-2|8-15|12-8|12-25|12-26| !semifeste|2015|12-24|
Focus:	Here are specified the days that are public Holydays this year, use the appropriate key: Holydays or Bank Holydays Note: configure here only the Holydays and Bank Holydays for which Planning is exceptional, without worrying about the ordinary Sundays. If your Planning includes a special Staff for Bank Holydays (for example: Friday before Easter, December 24), configure these days as Bank Holydays (do not specify Holydays and Bank Holydays if they have the same Staff requirements and Planning of the corresponding ordinary days, and therefore there is no Holydays and Bank Holydays Planning).
Detail:	Holidays (feste) and Bank Holydays (semifeste) of the year. **year: (AAAA)** **month: (MM)** **day: (DD)**
Error to avoid:	WARNING: configure here only the Holydays and Bank Holydays for which Planning is exceptional, without worrying about the ordinary Sundays. If your planning employs in Holydays and Bank Holydays the same staff employed in the common weeks and Sundays, do not configure anything.

	Configuration File Specifications **MANDATORY**
	Tag: !tipoturno
!tipoturno|*name-time1-time2-priority*|	
Example:	!tipoturno|Morning-8:00-13:30-0| !tipoturno|Afternoon-13:30-19:00-0| !tipoturno|Evening-19:00-22:00-1| !tipoturno|Night-22:00-06:00-3|
Focus:	Insert all Work Shifts that may be assigned to the Staff of your organization. At every Shift, assign a name to remember its features in the rest of the configuration. For example, a shop business could have the Shifts: * Early Morning (8:00 - 12:00) * Late Morning (9:00 - 13:00) * Early Afternoon (14:00 - 18:00) * etc.. But in an organization that works in continuous cycle could exist the Shifts: * Morning (6:00 - 14:00) * Afternoon (14:00 to 22:00 .) * Night (22:00 - 6:00) * Cleaning morning (06:00 - 08:00) * etc..
Detail:	One line for each kind shift that may be assigned to the staff of your organization. **name: (string maximum 40 characters long)** Name of the shift. i.e. Morning, Night, ShortShift. **time1: (HH:mm)** beginning time of the shift **time2: (HH:mm)** ending time of the shift **priority: (integer)** priority in the equable distribution. High priority must be used for unwanted shifts. i.e. nights, bank holidays and so on. It's value goes from 0 to 3: 0 lower priority (default) 1 normal priority 2 medium priority 3 highest priority
Error to avoid:	

	Configuration File Specifications **OPTIONAL**
	Tag: !squadra
!squadra|*team name-mandatory team|department|[letter-name|]*	
Example:	!squadra|Team one-0|Store|q-Vendor|q-Junior Vendor|u-Albert|u-Bob|u-Carl| !squadra|Team two-0|Store|q-Vendor|q-Junior Vendor|u-Albert|u-Ed|u-Fred| !squadra|Team two-0|Store|q-Vendor|q-Junior Vendor|u-Greg|u-Herbert|u-Leo|

Focus:	In a Department there may be the requirement that Persons (having one or more Qualifications) work as a Team. Configure the Teams taking into account Persons who may belong to more Teams, and considering that Persons not configured in a Team will be included in any Team (if Qualification matches). For example, if the Persons: Albert+Bob+Carl+Peter or Albert+Bob+Carl+Simon, or Albert+Ed+Fred+Peter or Albert+Ed+Fred+Simon, or Greg+Herbert+Leo+Peter or Greg+Herbert+Leo+Simon have to work together, then three Teams are to be set up: Albert+Bob+Carl, Albert+Ed+Fred, Greg+Herbert+Leo. Person Albert is specified because he can work in the first two Teams but not in the third one (and so Bob etc.), while the Persons Peter and Simon are not specified, because they can work in any Team.
Detail:	**team name: (string maximum 40 characters long)** Name you assigned to the team. **mandatory team: (boolean)** 0 not mandatory 1 for mandatory team **department: (string maximum 40 characters long)** Name of the department in which the team works. One of the department created before. **letter: (character)** q for qualification u for an employee **name: (string maximum 40 characters long)** if letter is q, this field is the name of one of the qualification created if letter is u, this field is the name of one employee specified created
Error to avoid:	

	Configuration File Specifications **MANDATORY**
	Tag: !reparti
!reparti|*name-flag-external department_beginning-external department end*|	
Example:	!reparti|Ambulatory-0--| !reparti|Travel-1-Manchester-Liverpool|
Focus:	In general, it does not make sense to use external departments for automatic scheduling, because they are useful only to account properly working hours, supplied on a loan to an entity external to the organization for unforecast needs. If you use the control of Cost by Goal (configuration of Persons), it may be necessary to have an external department in which manually move the shifts that the staff has done in external entities, because Persons were lent to them to make up for sudden needs. Work supplied in external departments has cost equal to zero in the processing phase: Reduction to Minimum Cost by Goal In this example |Travel-0-Manchester-Liverpool| means that you have a department called Travel that increments the working cost of the employee and that starts from Manchester and arrive to Liverpool. It maybe a bus shift.

Detail:	One line for each department. **name: (string maximum 40 characters long)** Name of the department **flag: (boolean)** Used for external departments. 0 – Increments the cost of the employee (i.e. the employee is loaned to another business). It's default value. 1 - doesn't increment the cost of the employee (i.e. the employee is loaned to another business) **external department beginning**: **(string maximum 40 characters long)** Used for travelling people. Depot where travel begins. **external department end**: **(string maximum 40 characters long)** Used for travelling people. Depot where travel ends
Error to avoid:	You need to enter at least one department to configure the schedule, even if the organization does not make use of the division into separate departments.

	Configuration File Specifications **MANDATORY**
	Tag: !qualifiche
!qualifiche|*name*|	
Example:	!qualifiche|Surgeon|
Focus:	Qualification (tasks) existing in Organization. Remember that every Person is entitled to work with a Qualification or more Qualifications, and in one Department or multiple Departments. So initially configure the existing Qualification in a general way, limiting to a few items. Then, setting up Persons, you can further subdivide the Qualifications in order to accurately represent the needs of your Planning.
Detail:	**name: (string maximum 40 characters long)** One line for each qualification.
Error to avoid:	You need to enter at least one professional qualification to configure the schedule, even if the organization does not make use of the division into separate professional qualifications.

	Configuration File Specifications **OPTIONAL**
	Tag: !depositi
!depositi|*name1*|*name2*|	
Example:	!depositi|Liverpool|Manchester|
Focus:	Only for organizations managing transport lines (pullman, truck, train). Insert every possible origin and destination (terminus) of your travels.
Detail:	Only for organizations managing transport lines (pullman, truck, train). One line for each origin or destination of travels. Insert every possible origin and destination (terminus) of your travels. **name1: (string maximum 40 characters long)** origin of the travel **name2: (string maximum 40 characters long)** destination of the travel

Error to avoid:	

	Configuration File Specifications **MANDATORY**
	Tag: !coprire
!coprire|*letter[-days of the week][-type of holiday][specific date1-specific date2]* *|department|qualification|shift name-num1-num2-num3-num4|*	
Example:	!coprire|s:MO-TU-WE-TH-FR-SA|Hospital|Surgeon|Morning-4-0-0-0| !coprire|t:fe|Hospital| Surgeon|Afternoon-3-1-1-0| !coprire|t:sf|Hospital|Surgeon|Night-3-1-1-0| !coprire|d:2015-01-19-2015-01-23|Hospital|Surgeon|Morning -1-0-0-0| !coprire|a:2015-01-19-2015-01-23|Hospital|Surgeon|Morning -1-0-0-0|
Focus:	Here you specify the Staff coverage requirements that will be NECESSARY and MANDATORY in the next automatic processing of Planning. The coverage of Shifts defined by those tags is applied in the processing of automated Planning, and to the FUTURE period for automatically Planning, whose you have configured the start and end dates. Plannings previously executed are stored and remain visible until they are deleted, but they are not modified in any way by changes in coverage requirements of Shifts.
Detail:	One line for each shift/qualification/department/range of date **letter: (character)** letter can be: s for standard weeks t for holidays (fe) and bank holidays (sf) d replaces for that department and qualification shift in a specific date a add people to a shift in a specific date **days of the week: (xx-xx-xx-xx-xx-xx-xx) only for letter s** days in which the shift is requested. You have to specify only the days you need. Each day is 2-characters long. **type of holiday: (2 characters) only for letter t** fe for holidays sf for bank holidays **specific date1: (aaaa-mm-gg) only for letter a or d** date of the beginning day **specific date2: (aaaa-mm-gg) only for letter a or d** date of the ending day if beginning and ending day are the same write only specific date1 **department : (string maximum 40 characters long)** name of one of the departments created **qualification : (string maximum 40 characters long)** name of one of the qualifications created **shift name: (string maximum 40 characters long)** name of one of the shifts created

	num1 : (integer) number of persons absolutely required in that shift **num2 : (integer)** number of persons optionally required in that shift (to add to num1) **num3 : (integer)** priority. 1 for shift with covering priority, 0 default no priority **num4 : (integer) default value 0** only for letter s: 1 for Copertura da riempire procedendo per settimane intere: l'elaborazione automatica della pianificazione procederà assegnando alle persone questo tipo di impegno per settimane intere, non per singoli turni o giorni. Questo parametro va quindi usato per pianificazioni che sono strettamente organizzate sulla base di un ritmo settimanale rigido e obbligatorio.
Error to avoid:	

Employment Contracts tags

Each employment contract has its properties that must be specified in order to obtain a real scheduling.

There many tag for a employment contract and they must be specified one after the other for each job contract.

Some are mandatory the others are optionally used for specific conditions.

The first one must always be the tag !contratto.

	Configuration File Specifications **MANDATORY**
	Tag: !contratto
!contratto|*name*|*field1*|*field2*|*field3*|*field4*|*field5*|*field6*| *field7*|*field8*| *field9*|	
Example:	!contratto|Employees|5:30|38:30-168:00|0:00-0:00|38:30-0:00|24:00-D6| 11:00|8:00-0:15|5:30-0|0:00-0:00|
Focus:	The Contracts express mandatory conditions to be complied with in relation to the Staff. For employees, insert conditions that correspond to the employment Contract legally binding. Sometimes the Shifts are covered by the owner or partners in the enterprise, in this case enter a Contract with their availability conditions (regardless of whether a legally valid contract exists). If an employee performs his work with special agreements, other than those applicable to others in his class, insert a special Contract that expresses the conditions that are applied effectively to this employee.
Detail:	**name: (string)** Name assigned to the job contract (string) **field1: (HH:mm)** Ordinary Hours of service in a day. When the commitment in a day exceeds this value, the excess Hours are considered Overtime, or recovery from previous Permits / Vacations / Absences. **field2: (HH:mm-HH:mm)** part1: Ordinary Hours of service during the week part2: Duration of the week (The week is normally seven days, but some Plannings may have limits similar to weekly ones over a shorter period for example, the standard service could be 30 Hours every six days) **field3: (HH:mm-HH:mm)** unused, for further parameters. Assign: 0:00-0:00 **field4: (HH:mm-HH:mm)** part1: maximum Hours of service that can be assigned in a week (including any Overtime or recovery) . It must be equal or greater than field2 part1 part2: unused set to 0:00 **field5: (HH:mm-D*number*)** part1: duration of the complete weekly break. Usually after 5 or 6 days on duty. part2: number of days before break. The week is normally 7 days, but some tiresome activities may require a closer constraint, such as 24 Hours of rest every 6 or 5 days, therefore after 5 or 4 days on duty **field6: (HH:mm)** Hours of service maximum that can be assigned on a day (including any Overtime or recovery).

	field7: (HH:mm-HH:mm) part1: Hours of minimum break between Shifts. This data is highly significant for Plannings involving alternating morning, afternoon and night Shifts. For example, if the minimum break is 11 Hours, and if a Person worked between 6:00 and 14:00, he cannot be engaged in the subsequent night Shift that begins at 22:00 (i.e., after only 8 Hours). part2: Break between shifts that is not considered as a break. For example: if you have a shift from 9 to 15 and another from 15.15 to 19, those 15 minutes are not calculated in the duration of the sheets. Usually you can set it to 0:15 **field8: (HH:mm-char)** Intra Moenia activities are typical of the Contracts of Health. A Person pays a part of his work time as a self employed professional, using the facilities of the institution. The Hours of Intra Moenia may be used by Planning to cover services for which the Employees are not sufficient; in this case Intra Moenia activities are relevant to Planning, and must be configured. Otherwise set this field to 0:0-0 part1: Maximum daily hours for Intra Moenia activities part2: '0' for days with only Intra Moenia activities are not allowed '1' for days with only Intra Moenia activities are allowed Usually Intra Moenia activities are permitted only for a few Hours before or after a Shift provided by the Person as an employee of the institution. Enable this check if Planning needs require that sometimes some Persons can only be used in Intra Moenia activities. **field9: (HH:mm-HH:mm)** These configuration data have meaning only if planning can be composed of more than one shift in a day, and every shift is much shorter than the working day. Otherwise set it to 0:0-0:0 For example, if exists a morning shift of 4 hours and a noon shift of 2 hours, and if it is allowed that a work day contain 4 or 4+more hours with a maximum of 7, but not only 2 hours, then the minimum working time per day will be configured equal to 4 hours. More, if exists also an afternoon shift of 3 hours, you can configure the vacant time to be avoided to 2 hours, in order to avoid that a person can have the morning and the afternoon shift in the same day, with a vacant time equal to the noon shift, assigned to some other person. The Minimum working time per day is a mandatory constraint and is respected exactly. The Vacant time to be avoided is respected approximately in the processing phase: Vacant times among short Shifts. part1:Minimum working time a day when the organization is based on multiple short shifts part2:Vacant time to be avoided between shifts in the same day
Error to avoid:	

	Configuration File Specifications **OPTIONAL**
	Tag: !copausapranzo
!copausapranzo|[*starting hour-ending hour*|]	

Example:	!copausapranzo|12:30-15:30| !copausapranzo|12:30-13:30|13:00-14:00|
Focus:	The lunch break allows the interruption of the activity of a working day without applying the minimum break required between Shifts. For example, if a business provides morning or afternoon Shifts, lunch breaks can be set between 12:00 and 16:00. If an activity involves continuous Shifts of 7 or 8 Hours, lunch break is not to be set up (after each Shift the requirement of minimum break must be met).
Detail:	**starting hour: (HH:mm)** starting hour of the break **ending hour: (HH:mm)** ending hour of the break Maximum 4 breaks in the command
Error to avoid:	

	Configuration File Specifications **MANDATORY**
	Tag: !costraordinario
!costraordinario|*maximum hours a year-maximum hours a month|cost of overtime|special starting hour-special ending hour-starting day-ending day*|	
Example:	!costraordinario|0:00-0:00|200|0:00-0:00-0-0| !costraordinario|288:00-24:00|200|20:00-6:00-5-0|
Focus:	Overtime Hours maximum payable in the month and year. The Overtime paid beyond those limits is not permitted. Those within these limits are allowed, but the calculation of Planning will reduce the use to a minimum, and if possible to avoid at all.
Detail:	**maximum hours a year: (HH:mm)** maximum number of hours/minutes of overtime permitted in a year **maximum hours a month: (HH:mm)** maximum number of hours/minutes of overtime permitted in a month **cost of overtime: (integer)** Cost per Hour of Overtime, where the normal engagements have cost = 100. As the use of Overtime should be reduced to a minimum, you should always set a cost for Overtime that is greater than the cost of ordinary Hours used (= 100) and also bigger than the cost of Hours spent in Substitutions. Initially assign the cost 150 to Substitutions and 200 to Overtime. Later, you can adjust this value in order to use Overtime or Substitution in the optimal proportion. **special starting hour: (HH:mm)** some hours in specific days could be considered as overtime. e.g. the nights, or weekends. Here you can specify the starting hour of the period to be calculated as l overtime. **special ending hour: (HH:mm)** Here you can specify the ending hour of the period to be calculated as overtime . **starting day: (integer)** number of the starting day of the week:

	0 Monday … 6 Sunday **ending day: (integer)** number of the ending day of the week: 0 Monday … 6 Sunday
Error to avoid:	The field is mandatory but if you don't want to use overtime put this tag with default values: !costraordinario|0:00-0:00|200|0:00-0:00-0-0|

	Configuration File Specifications **MANDATORY**
	Tag: !conotturno
!conotturno|*starting night hour-ending night hour*|*maximum length*-24:00|*num1-break*|	
Example:	!conotturno|0:00-0:00|0:00-24:00|0-0:00| !conotturno|20:00-6:00|8:00-24:00|3-24:00|
Focus:	A Shift is considered Nocturnal if it is wholly or partly within the limits set up here. Caution: for example, if there were only Shifts 06:00-14:00, 14:00-22:00, 22:00-6:00, the limits on night work would obviously be 22:00 to 6:00. But if there was also an evening Shift 15:00-23:00, and this should not be considered nocturnal, then the night the interval should be set up as 23:00-6:00. Shifts 22:00 to 6:00 would be considered nocturnal, because they are within this range (even if only PARTIALLY). Maximum length of night Shifts. Configure this with lower values than those of the day if, for example, night Shifts do not allow Overtime, or admit it to a limited extent. In addition, this data is useful in cases where there are several types of night Shift, short, which could be allocated following one another for the same Person (e.g. from 22:00 to 01:00 a task, from 1:00 to 4:00 another task, etc.).
Detail:	**starting night hour: (HH:mm)** Hour to consider as beginning of night work. **ending night hour: (HH:mm)** Hour to consider as ending of night work. **maximum lenght: (HH:mm)** Maximum length of night Shifts. Configure this with lower values than those of the day if, for example, night Shifts do not allow Overtime, or admit it to a limited extent. In addition, this data is useful in cases where there are several types of night Shift, short, which could be allocated following one another for the same Person (e.g. from 22:00 to 01:00 a task, from 1:00 to 4:00 another task, etc.). **num1: (integer)** Maximum number of consecutive nocturnal shifts allowed. **break: (HH:mm)** Hours of rest required after a series of night shifts.
Error to avoid:	The field is mandatory but if you don't want to use nocturnal work put this tag with default values: !conotturno|0:00-0:00|0:00-24:00|0-0:00|

	Configuration File Specifications **MANDATORY**
	Tag: !codovuto
!codovuto|*number of working hours-number of vacation hours*|*substitution cost*|*number of Intramoenia hours*-100|	
Example:	!codovuto|1748:00-270:00|150|800:00-100| !codovuto|2080:00-280:00|150|0:00-100|
Focus:	Total Hours of work to be provided in the year. For example: totally, 40 Hours * 52 weeks = 2080 Hours. Vacations and Permits: 35 days, it is 8 Hours for 35 days = 280 Hours. Hours to be worked = 2080-280 = 1800. Enter this information carefully in order to calculate correctly the engagement of Staff members and to better approximate the ideal target (which is to use all the Staff exactly to 100% of the Hours to be provided, without the use of Overtime). Do not specify this data for the self employed consultants, substitutes, etc. that are called into service only when the Permanent Staff is not sufficient.
Detail:	**number of working hours: (HH:mm)** Total Hours of work to be provided in the year **number of vacation hours: (HH:mm)** Total Hours of Vacations and Permits to be given in the year **substitution cost: (integer)** Cost per Hour of Substitution engagements, where the normal engagements have cost = 100. The Substitution engagements are those in which a Person is assigned to a Qualifications that is lower of his professional degreem, in order to substitute other not available Persons (because absent, or already used to the maximum Contract Hours). Since the Shifts in Substitution should be kept to a minimum, you should always set a cost higher than 100 for the Substitutions. Initially assign the cost 150 to Substitutions and 200 to Overtime. Later, you can adjust this value in order to use Overtime or Substitution in the optimal proportion. **number of Intramoenia working hours: (HH:mm)** Total Hours of Intramoenia work to be provided in the year
Error to avoid:	The field is mandatory but if you don't want to use this options put this tag with default values: !codovuto|0:00-0:00|150|0:00-100|

	Configuration File Specifications **OPTIONAL**
	Tag: !cowke
!cowke||*starting we hour-ending we hour*|*starting day-ending day*|*maximum we*|	
Example:	!cowke|20:00-6:00-4-0|2|
Focus:	Weekend time limits. If this data is configured, the processing tries to distribute approximately the Weekend commitments in the most equitable way, in the processing phase: Equable distribution of Free Weekends and Holydays. The constraint of Maximum consecutive Weekends with commitments is mandatory and is respected exactly.

Detail:	**starting we hour: (HH:mm)** Hour to consider as beginning of the weekend. **ending we hour: (HH:mm)** Hour to consider as ending of the weekend. **starting day: (integer)** number of the starting day of the weekend: 0 Monday … 6 Sunday **ending day: (integer)** number of the ending day of the weekend: 0 Monday … 6 Sunday **maximum we: (integer)** maximum consecutive weekends with commitments.
Error to avoid:	

	Configuration File Specifications **OPTIONAL**
	Tag: !coorvc
!coorvc|*starting hour-ending hour-percentage1-percentage2-percentage3-percentage4-percentage5-flag*|	
Example:	!coorvc|22:00-6:00-10-5-10-10-5-1|
Focus:	Accountable value of hours - surcharges. Hours worked in the night or on holydays can have a percentual surcharge of their accountable value. Surcharges for holidays are applied choosing always the biggest among them; for example, if common Sundays have a surcharge of 20% and holydays like Christmas have a 40% surcharge, and if in a given year Christmas is on Sunday, 40% surcharge is applied. If nocturnal and holyday surcharge are both to be applied, you can configure whether to apply both or choose the biggest one.
Detail:	**starting hour: (HH:mm)** Hour to consider as beginning of the night period with surcharge. **ending hour: (HH:mm)** Hour to consider as ending of the night period with surcharge. **percentage1: (integer)** Percentage of cost surcharge for nights. **percentage2: (integer)** Percentage of cost surcharge for Saturdays. **percentage3: (integer)** Percentage of cost surcharge for Sundays. **percentage4: (integer)** Percentage of cost surcharge holidays. **percentage5: (integer)**

	Percentage of cost surcharge bank holidays. **flag: (integer)** If nocturnal and holyday surcharge are both to be applied, you can configure whether to apply both (assign value 1) or choose the biggest one (assign value 0).
Error to avoid:	

	Configuration File Specifications	**OPTIONAL**
	Tag: !coorpa	
Example:		
Focus:	Unused, for further expansion	
Detail:		
Error to avoid:		

	Configuration File Specifications	**OPTIONAL**
	Tag: !coorpb	
Example:		
Focus:	Unused, for further expansion	
Detail:		
Error to avoid:		

	Configuration File Specifications	**OPTIONAL**
	Tag: !comwl	
Example:		
Focus:	Unused, for further expansion	
Detail:		
Error to avoid:		

Employees tags

Each employee has his properties that must be specified in order to obtain a real scheduling.

There many tag for an employee and they must be specified one after the other for each person.

Some are mandatory the others are optionally used for specific conditions.

The first one must always be the tag !u.

	Configuration File Specifications **MANDATORY**
	Tag: !u
!u|*name of the employee*|0|	
Example:	!u|Billund|0|
Focus:	Every Person can be used in one or more Departments, with one or more Qualifications. Specify all the possibilities. It may be that a Person can be used with a specific Qualification in any Department: in this case you can use the option "Any Department" and not to enumerate all the different Departments. In theory, a Person may also be used with any Qualification (this case, however, is not likely in the reality). People can be used in certain Departments and with certain Qualifications as ORDINARY work (because it corresponds to the professional Qualification of the Person) or in Substitution (for Staff Persons absent or already committed to the maximum allowable). The use of the Persons in Substitution has a higher cost (configured in the Contract) and is avoided as much as possible. Substitution NOT PREFERRED: is used only if a given shift cannot be assigned as Ordinary work or simple substitution.
Detail:	One line for each person. **name: (string maximum 40 characters long)** Name assigned to the employee **flag**: **(boolean)** Unused, assign 0.
Error to avoid:	

	Configuration File Specifications **MANDATORY**
	Tag: !uco
!uco|*name of the contract-num1-num2*|	
Example:	!uco|Employees-0-0| !uco|Employees with reduced annual hours-0-0|
Focus:	Each tag !u must be followed by a tag !uco. This tag specifies the job contract of the employee.

Detail:	**name of contract: (string maximum 40 characters long)** Name of one of the job contract created **num1**: **(integer)** Unused, assign 0. **num2**: **(integer)** This number represents the Control of Cost by Goal. 0 means that the specified employee has cost proportional to the work done (default), the control of the Cost by Goal is not active. 1 is used in the case that some members of the staff (such as managers or apprentices) have a fixed conventional monthly cost, while others have cost in proportion to the hours of work. If at least one staff member is configured with conventional fixed monthly cost, the automatic processing of scheduling plays an optimization phase of Cost by Goal (Reduction to Minimum Cost by Goal), which reduces as much as possible the commitment of the staff whose cost is proportional to the worked time.
Error to avoid:	

	Configuration File Specifications **MANDATORY**
	Tag: !ustoria
!ustoria|0:00|*total annual hours*|0:00|*total annual vacation hours*|0:00|*total annual Intramoenia hours*|	
Example:	!ustoria|0:00|0:00|0:00|0:00|0:00|
Focus:	Each tag !u must be followed by a tag !uco and a tag !ustoria. This tag specifies the annual hours of the employees: job and vacation.
Detail:	**total annual hours: (HH:mm)** total annual hours of work. This number represents the total number of hours worked by the employee until the previous scheduling specified by the tag !uimpo (excluded). e.g. **total annual vacation hours: (HH:mm)** total annual hours of vacations used by the employee. This number represents the total number of vacation hours used by the employee until the previous scheduling specified by the tag !uimpo. e.g. **total annual hours: (HH:mm)** total annual hours of work. This number represents the total number of hours worked as Intramoenia by the employee until the previous scheduling specified by the tag !uimpo (excluded).
Error to avoid:	The field is mandatory but if you don't want to use overtime put this tag with default values: !ustoria|0:00|0:00|0:00|0:00|0:00|0:00|0:00|

	Configuration File Specifications **OPTIONAL**
	Tag: !ucondizioni

!ucondizioni|*num1-work hours*|*vacation hours*|*Intramoenia hours*|	
Example:	!ucondizioni|59-0:00|0:00|0:00| !ucondizioni|59-1400:00|120:00|300:00|
Focus:	
Detail:	**num1: (integer)** start date of employment, expressed as absolute day of current year, where January 1th = 0 and February 1th= 31 and so on. If a Person begins service in the current year, enter the exact start date of the service, in order to correctly calculate his working Hours / Vacations to be granted. If a Person has started service before the start of the current year, the start date must be January 1. CAUTION: if a Person is temporarily employed, e.g. from 1 July to 30 September, the Hours of work to be done will be one quarter of the yearly amount. But the the program will show the Hours of half an year, and this should NOT be correct: the fact that the Person will cease the service before the year's end is not relevant to the Planning of July, August and September, because the program will seek to engage the Person at 100% of the Hours to be paid for each month, and the planned termination of service on September 30 has no relevance. **work hours: (HH:mm)** Total Hours of work to be paid in the current year, related to the actual start date of employment (equal to the yearly total if the Person is in service since January 1). If 0:00, the Server calculates automatically the number of hours depending on the job contract. **vacation hours: (HH:mm)** Total Hours of vacation to be paid in the current year, related to the actual start date of employment (equal to the yearly total if the Person is in service since January 1). If 0:00, the Server calculates automatically the number of hours depending on the job contract. **Intramoenia hours: (HH:mm)** Total Hours of Intramoenia to be paid in the current year, related to the actual start date of employment (equal to the yearly total if the Person is in service since January 1). If 0:00, the Server calculates automatically the number of hours depending on the job contract.
Error to avoid:	

	Configuration File Specifications **OPTIONAL**
	Tag: !uprefo
!uprefo|*letter-shift-perc1-perc2-flag*| !uprefo|[*starting hour-ending hour-perc1-perc2-flag*|] *(maximum 4 periods)*	
Example:	!uprefo|t-Morning-40-60-0|t-Afternoon-30-70-0| !uprefo|8:00-16:00-60-80-0|22:00-6:00-0-0-1|
Focus:	Use this configuration to express the preference for certain types of Shifts or time brackets. A Person can express the requirement, for example, to be engaged in the morning to up to 60% of his work time but at least 40%, or being engaged in the Afternoon at least 30%, and so on. If a Person can NEVER

	be committed in a given time slot, do not set the constraint here, but the in configuration of the Vacations, leaves and times not available.
Detail:	**letter: (character) not mandatory** if it's specified (its value can be only **t**) than it's followed by a shift **shift: (string maximum 40 characters long)** name of one of the shifts created **starting hour: (HH:mm)** Hour to consider as beginning of the period. **ending hour: (HH:mm)** Hour to consider as ending of the period. **percentage1: (integer)** Minimum percentage of engagement in the shift specified **percentage2: (integer)** Maximum percentage of engagement in the shift specified **flag: (boolean)** If a Person can NEVER be committed in a given time slot, set **1** (0-0-1), otherwise set **0** (40-60-0)
Error to avoid:	

	Configuration File Specifications **OPTIONAL**
	Tag: !uprefg
!uprefg|[*day of week-minimum perc-maximum perc-flag|]* *(maximum 7 days)*	
Example:	!uprefg|0-0-0-1|1-0-20-0|2-0-20-0|3-0-20-0|4-0-20-0|5-20-40-0|6-20-40-0|
Focus:	Use this configuration to express a preference for certain days of the week. A Person can express the requirement, such as not to be committed on Monday or being engaged on Sunday at least 20% but up to 40%, and so on. In this example there's a person who wants to work more in the weekend, less in the other days, never on Monday.
Detail:	**day of the week: (integer)** number of the specific day of the week: 0 Monday … 6 Sunday **minimum perc: (integer)** Minimum percentage of engagement in the day specified **maximum perc: (integer)** Minimum percentage of engagement in the day specified **flag: (boolean)** If a Person can NEVER be committed in a given day, set **1** (0-0-1), otherwise set **0** (20-40-0)
Error to avoid:	

	Configuration File Specifications **OPTIONAL**

	Tag: !upremored
!upremored|[*day of week-maximum minutes|] (maximum 7 days)*	
Example:	!upremored|1-240|2-240|3-240|4-240|5-480|6-480|
Focus:	This tag is used to specify an exact working period (expressed in minutes) in one or more specific days.
Detail:	**day of the week: (integer)** number of the specific day of the week: 0 Monday … 6 Sunday **maximum minutes: (integer)** Maximum number of minutes of engagement in the day specified
Error to avoid:	

	Configuration File Specifications **OPTIONAL**
	Tag: !uprero
!uprero|*letter-shift-minimum number-maximum number-starting day-flag exact-flag pause*| !uprero|[*starting hour-ending hour-minimum number-maximum number-starting day-flag exact-flag pause |] (maximum 4 periods for each !uprero)*	
Example:	!uprero|t-Morning-3-4-5-0-1| !uprero|8:00-16:00-2-4-0-0-0|
Focus:	Use this configuration to express the event that certain employees must be committed with a constant alternating rhythm, for example: 2 days in the morning, 2 in the afternoon. Or for the case where a certain type of shift should be grouped in sequences (e.g., the shift "Kitchen" must be assigned in sequences long from 3 to 5 days). Exact limits: if the exact limits check is selected, the shift sequence is respected only if shifts match exactly with the configured limits; else it is considered respected also if the shifts match partially with the configured limits.
Detail:	**letter: (character) not mandatory** if it's specified (its value can be only **t**) than it's followed by a shift **shift: (string maximum 40 characters long)** name of one of the shifts created **starting hour: (HH:mm)** Hour to consider as beginning of the period. **ending hour: (HH:mm)** Hour to consider as ending of the period. **minimum number: (integer)** Minimum number of consecutive shifts of the kind specified **maximum number: (integer)** Maximum number of consecutive shifts of the kind specified **flag exact: (boolean)** if **1** the shift sequence is respected only if shifts match exactly with the configured limits; if **0** it is considered respected also if the shifts match partially with the

	configured limits. **flag pause: (boolean)** used in case for vacation. if **1** the shift sequence must be continued after the vacation if **0** there will be a new beginning of the sequence.
Error to avoid:	

	Configuration File Specifications **OPTIONAL**
	Tag: !upretuo
!upretuo|t-*department-quailfication-shift-minimum sequence number-maximum sequence number-pause-maximum times-number of days-flag-new shift-flag exact*|	
Example:	!upretuo|t--Sugeon-Afternoon-0-0-0-0-0-0-Morning-0| !upretuo|t---Night-0-0-0-0-0-0-Afternoon-0| !upretuo|t-L101--Night -0-3-7-0-0-0-Afternoon-0| !upretuo|t-L107--Night -0-3-7-0-0-0-Afternoon-0|
Focus:	This data are useful to fairly distribute the tasks that the staff usually likes less. Use these fields to configure limitations to the shifts in a given [department] and/or [qualification] and/or [shift type]. Each data is optional, but at least one should be specified. Than you can configure (all is optional): * the minimum or maximum of consecutive days in which you can assign the person to the given department/qualification/shift type, and/or * the maximum of shifts in the given department/qualification/shift type which can be given in a period of n days, calculated on the base of the average in the planning period, and/or * the pause in days that must be observed before coming back to the the given department/qualification/shift type. Configure only the data necessary in order to obtain a good planning, without redundancies.
Detail:	**department : (string maximum 40 characters long)** name of one of the departments created. Optional, if not specified, it means any department **qualification : (string maximum 40 characters long)** name of one of the qualifications created. Optional, if not specified, it means any qualification **shift: (string maximum 40 characters long)** name of one of the shifts created. Optional, if not specified, it means any department **minimum sequence number: (integer)** Minimum number of consecutive days in which you can assign the person to the given department/qualification/shift type **maximum sequence number: (integer)** Maximum number of consecutive days in which you can assign the person to the given department/qualification/shift type **pause: (integer)** the pause (number of days) that must be observed before coming back to the the given department/qualification/shift type.

	maximum times: (integer) Maximum number of sequence in number of days specified in the next field **number of days: (integer)** Number of days to calculate the number of sequences. e.g. maximum 3 times on 14 days **flag: (boolean)** if **1** maximum sequence number =0 if **0** otherwise **new shift: (string maximum 40 characters long)** name of one of the shifts created. It's the starting shift required after the sequence. **flag exact: (boolean)** if **1** the shift sequence is respected only if shifts match exactly with the department, qualification and kind of shift. if **0** it is considered respected also if the shifts match partially with with the department, qualification and kind of shift.
Error to avoid:	

	Configuration File Specifications
	Tag: !ulimtutor
Example:	
Focus:	
Detail:	Unused
Error to avoid:	

	Configuration File Specifications **OPTIONAL**
	Tag: !ulimorvie
!ulimorvie|t-*department-qualification-starting hour-ending hour-MTWTFSSHB*|	
Example:	!ulimorvie|t-Hospital-Surgeon-0:00-0:00-010000010|
Focus:	This configuration should be used in the case that a person cannot be engaged: * in a given department, or * with a given qualification, or * in a given department and with a given qualification in the specified time bracket and/or in the specified days of the weeks (and/or holydays).
Detail:	**department : (string maximum 40 characters long)** name of one of the departments created. Optional, if not specified, it means any department **qualification : (string maximum 40 characters long)** name of one of the qualifications created. Optional, if not specified, it means any qualification

	starting hour: (HH:mm) Beginning time of a period in which a person cannot be engaged. If not necessary set to 0:00 **ending hour: (HH:mm)** Ending time of a period in which a person cannot be engaged If not necessary set to 0:00 The flags below are used to specify that a person cannot be engaged in a day of the week or in an holiday. **M: (boolean)** For Monday. 0 can be engaged, 1 cannot be engaged **T: (boolean)** For Tuesday. 0 can be engaged, 1 cannot be engaged **W: (boolean)** For Wednesday. 0 can be engaged, 1 cannot be engaged **T: (boolean)** For Thursday. 0 can be engaged, 1 cannot be engaged **F: (boolean)** For Friday. 0 can be engaged, 1 cannot be engaged **S: (boolean)** For Saturday. 0 can be engaged, 1 cannot be engaged **S: (boolean)** For Sunday. 0 can be engaged, 1 cannot be engaged **H: (boolean)** For Holidays. 0 can be engaged, 1 cannot be engaged **B: (boolean)** For Bank Holidays. 0 can be engaged, 1 cannot be engaged
Error to avoid:	

	Configuration File Specifications **OPTIONAL**
	Tag: !uviaggia
!uviaggia|*depot-starting time-maximum time outside-time gap*|	
Example:	!uviaggia|ancona-2880-240-60|
Focus:	If the person works in departments corresponding to transport lines, insert: * the depot by which the person has his domicile, where he must start from and go back after a sequence of shifts; * the idle time after which the person must start from his domicile (and before which he must go back to it); * the maximum idle time allowed when the person is outside his domicile; * the time between shifts when the person is outside his domicile (can be zero if restarting soon is allowed.)
Detail:	**depot : (string maximum 40 characters long)** name of the depot by which the person has his domicile, where he must start from and go back after a sequence of shifts **starting time: (integer)** the idle time after which the person must start from his domicile and before which he must go back to it **maximum time outside: (integer)** the maximum idle time allowed when the person is outside his domicile

	time gap: (integer) the time between shifts when the person is outside his domicile, can be zero if restarting soon is allowed
Error to avoid:	

	Configuration File Specifications **OPTIONAL**
	Tag: !uprevarie
!uprevarie|*number*|	
Example:	!uprevarie|8|
Focus:	This constraint is to be used in cases in which planning is based essentially on the periodic sequence of rest times. For example, if planning is based on six work days and two rest days, and it is necessary to guarantee that the next rests begins always exactly eight days after the previous one, configure the value 8.
Detail:	**number: (integer)** number of days between rests
Error to avoid:	

	Configuration File Specifications **CREATED BY SERVER**
	Tag: !uimpo
!uimpo|*flag1-department-qualification-[*T*datetime:duration]*|	
Example:	!uimpo|04-External Unit-Surgeon-T201501020800:330| !uimpo|00-Hospital-Surgeon-T201501021330:330-T201501030800:330-T201501031330:330-T201501050800:330| (no limit for number of periods)
Focus:	This tag represents the various engagements of the current employee in the current scheduling and in the past ones too. So there plenty of uimpo tag for each employee.
Detail:	**flag1 : (string)** **00 –** Ordinary work **04 -** Intramoenia **10 -** Vacation or permits enjoyed (increases the Vacations account) **20** - Sickness or other paid absence (increases the Work account) **40** - Preferred free time (it does not account either as Vacation or as Work) **80 –** Engaged in a department not in planning (increases the Work account) **department : (string maximum 40 characters long)** name of one of the departments created. Optional, if not specified, it means any department **qualification : (string maximum 40 characters long)** name of one of the qualifications created. Optional, if not specified, it means any qualification **datetime : (aaaammgghhmm)** starting date/time of the engagement

	duration : (integer) expressed in minutes
Error to avoid:	Riflettere quelli della frontiera

	Configuration File Specifications **OPTIONAL**
	Tag: !uferpe
!uferpe|d*flag:*Tstart*datetime:enddatetime*| !uferpe|s*flag:day of the week*|*start:end* !uferpe|t*flag:type*|*start:end*|	
Example:	!uferpe|d10:T201501160000:T201501170000| !uferpe|s40:MO|0:0| !uferpe|t40:fe|0:0|
Focus:	Use this page to configure the planned absences of Staff. Absences may be constant constraints (for example, a Person never available on Tuesday mornings), or can express Permits / Vacations enjoyed on specific dates. You can insert for memory also the Vacations planned for dates after the Automated Planning Period (for example, if we are in March and we are planning April, we can add also the Vacations already scheduled for June, July, etc.). The absence following the Automated Planning Period are stored, but have no effect on the Process of the Schedule. WARNING: Remember that the Vacations and absences configured FOR SPECIFIED PERIODS are used for the calculation of Vacations actually enjoyed in relation to the Contract conditions. That is, if for example you configure that an employee is never present on Tuesday morning, this does not affect his total of enjoyed Vacations, but if you configure that he will absent on Tuesday, April 3, this will determine the calculation of 8 Hour Permit as enjoyed (or better, of the ordinary Hours per day according to the Person's Contract).
Detail:	**flag: (string)** **10 -** Vacation or permits enjoyed (increases the Vacations account) **20** - Sickness or other paid absence (increases the Work account) **40** - Preferred free time (it does not account either as Vacation or as Work) **80 –** Engaged in a department not in planning (increases the Work account) **start datetime : (aaaammgghhmm)** starting date/time of the vacation period **end datetime : (aaaammgghhmm)** ending date/time of the vacation period **day of the week: (2 characters)** MO for Monday, TU for Tuesday and so on. **start: (integer)** starting minute in the day **end: (integer)** ending minute in the day if you set 0:0 it represents all the day **type: (string)** **fe:** for holidays **sf:** for bank holidays

Error to avoid:	

	Configuration File Specifications **OPTIONAL**
	Tag: !uperxe
!uperxe|*flag*-d*:*Tstart*datetime:enddatetime*| !uperxe|*flag*-s*:day of the week|start:end* !uperxe|*flag*-t*:type|start:end*|	
Example:	!uperxe|0-d:T201501011800:T201501012300| !uperxe|0-s:MO|0:0| !uperxe|1-s:TU|0:0|
Focus:	Use this tag to configure the time intervals in which Staff CAN be employed (for Persons whose availability is limited to certain days or periods, typically part-time or external consultants), or in which it MUST be employed having expressed the request. These requirements may result in constant constraints (for example, a part-time Person available only on Mondays and Tuesday mornings), or correspond to specific dates. You can also insert for memory requests scheduled for dates after the Automated Processing Period (for example, if we are in March and we are planning April, but a Person told us that between 5 and 10 June he wants to be on duty every day, we can insert this constraint already planned for June, July, etc.). Subsequent requests to the Automated Planning Period are stored, but have no effect on the processing of the Scheduler. WARNING: remember that the configurations available on this page are used ONLY to express the condition of non-availability of a Person in the Planning, and have no meaning for the calculation of Vacations actually enjoyed in relation to the Contract conditions.
Detail:	**flag: (boolean)** These two cases are possible: Restrictive: a Person CAN be engaged only in the available intervals configured. If you configure restrictions on availability, the Person may be committed only during periods set up here. The periods can be configured for different types, as necessary (for example, you can express that a Person is always available on Tuesday, on Wednesday afternoon, Thursday, on April 3 in the morning, etc.) Mandatory: the Person MUST be engaged in this range. If you configure mandatory periods, the Person must be engaged in all expressed periods, and may also be committed at any other time when not on Holyday or permit. Of course, you can completely configure the wider time frame in which a Person CAN be committed, and the smaller one, inside the first, in which a Person MUST be committed. **0 -** Restrictive: a Person CAN be engaged only in the available intervals configured **1 -** Mandatory: the Person MUST be engaged in this range **start datetime : (aaaammgghhmm)** starting date/time of the engaged period **end datetime : (aaaammgghhmm)** ending date/time of the engaged period **day of the week: (2 characters)**

	MO for Monday, TU for Tuesday and so on. **start: (integer)** starting minute in the day **end: (integer)** ending minute in the day if you set 0:0 it represents all the day **type: (string)** **fe:** for holidays **sf:** for bank holidays
Error to avoid:	

	Configuration File Specifications **MANDATORY**
	Tag: !urq
!urq|*department-qualification-num*|	
Example:	!urq|Hospital-Surgeon-0| !urq|External Unit-Surgeon-2|
Focus:	Every Person can be used in one or more Departments, with one or more Qualifications. Specify all the possibilities. It may be that a Person can be used with a specific Qualification in any Department: in this case you can use the option "Any Department" and not to enumerate all the different Departments. In theory, a Person may also be used with any Qualification (this case, however, is not likely in the reality). People can be used in certain Departments and with certain Qualifications as ORDINARY work (because it corresponds to the professional Qualification of the Person) or in Substitution (for Staff Persons absent or already committed to the maximum allowable). The use of the Persons in Substitution has a higher cost (configured in the Contract) and is avoided as much as possible. Substitution NOT PREFERRED: is used only if a given shift cannot be assigned as Ordinary work or simple substitution.
Detail:	**department : (string maximum 40 characters long)** name of one of the departments created. Optional, if not specified, it means any department **qualification : (string maximum 40 characters long)** name of one of the qualifications created. Optional, if not specified, it means any qualification **number: (integer)** 0 – ordinary 1 – substitution 2 – Intramoenia 3 – substitution not preferred
Error to avoid:	

PLANNING DIFFERENT PERIODS OF SCHEDULING

During the planning of a scheduling you have two periods to consider:
the actual period to plan
the previous period already planned.

When you create a new file for a new scheduling, the Server Application needs to use the pervious planning in order to calculate annual work hours, vacations, sequences of shifts and so on.
The Server Application returns you, for every scheduling, many lines with tag !uimpo in which are specified the engagements of employees.
When you create a new scheduling you must give the Server Application all those tags !uimpo of the previous planning.

Periodically, several times during each year, it’s better to run a Closing to delete the data in recent months and make management of the data less laborious.
When you want to close a selected period, the Hours of paid work (and holydays) are added to the Person's annual totalizer, and individual commitments for that period (tags !uimpo) of Persons must be deleted.
The time interval that can be closed must end at least 28 days before the beginning of the Actual Planning Period. Shifts committed in the 28 days prior to the Automated Planning Period must always fully kept because they are used to calculate the errors in the new period.

Early in every year (generally in one of the first days of January), run the Closing of the previous year. The total amount of worked hours (and holydays) in the current year is deleted and summed to the previous year’s total. In this way the calculation of due hours of work and holyday is reset for the new year.
Note: the start date of a year relevant for the calculation of working Hours to be granted is not necessarily January 1 - it could be, for example, the first Monday of the year, or the last Monday of the previous year. It depends on how you want to calculate the progressive total of worked Hours of the year, which could be calculated from January 1 to December 31, for example, or for 52 weeks, from first Monday of the old year until the first Sunday of the new year, or other such criteria.

DATA TO COLLECT BEFORE CREATING A CONFIGURATION

Before using ZonaTEAM it's necessary to collect some information that are mandatory. Here is a first form to fill with the client with basic elements to create a scheduling.

General Data:	Name of each office of the organization	
	Name of each department	
	Name of each qualification for the employees	
For each contract:	Number of ordinary working hours for each day	
	Maximum number of working hours including overtime and recovery for each day	
	Maximum Number of break hours between shifts	
	Number of ordinary working hours for each week	
	Maximum number of working hours including overtime and recovery for each week	
	Number of hours for a complete break, and after which number of days. (Usually 24 hours after 6 days)	
For each employee:	Name	
	Job contract (chose one between those specified before)	
	Main qualifications	
	Qualification permitted in substitution	
	Number of holiday hours	
For each shift:	Name	
	Starting hour	
	Ending hour	
	Days of the week for the shift	
	Priority. Chose the unwanded shifts, because they must be equally distributed between employees. i.e. Sunday, nights.	
	Number of employees absolutely necessary to cover the shift	
	Number of employees optionally added to the shift.	