

WINGS ACCOUNTING INTERFACE

(VERSION 7.95)

GENERAL

The Wings Accounting Interface (WAI) is conceived as a COM Automation Server. This implies that it supports dual interfaces, in other words both early and late binding of properties and methods are possible.

The link to WAI can be used in different ways. For example, the level of interfacing can range from a very tight integration where you always keep all files synchronized, to a batch processing of all record changes and bookings at regular time intervals. Furthermore, different levels of detail are possible when passing data, because WAI can assume default values for unused properties (see also: WAI.INI).

WAI supports the import of customers, suppliers, analytical accounts and bookings. Other data that is used with bookings, such as foreign currencies, general accounts and VAT codes are assumed to be present in Wings. Retrieving information (such as outstanding balances) from the accounting system through this interface is possible as well.

To avoid conflicts when importing data, Wings provides some extra security settings:

CUSTOMERS/SUPPLIERS

Per file you can opt for one of the following systems:

- Low numbering:

Wings uses the numbering range from 1 to 49999999. External programs use numbers from 50000000 on to avoid double use of a number.

Remark: Searching customers/suppliers by number is much easier when the numbers are low because the user only has to type the significant digits.

- High numbering:

Wings uses the numbering range from 50000000 to 99999999. External programs use numbers up to 49999999 to avoid double use of a number.

- Modification blocking:

The external customer/supplier file is the reference file. All additions and modifications have to be performed in the external program that takes care of the automatic synchronization of the files in Wings.

JOURNALS

It is possible to block all usage of a specific journal in Wings to make it uniquely accessible for the external program.

OBJECTS

The Wings Accounting Interface contains following COM objects:

YCONNECTION

Establishes or releases the connection with a Wings dossier. This object is needed to give the correct dossier context for all further actions.

YDOSSIER

Dossier object. Allows the creation of a new Wings dossier.

YCUSTOMER

Customer object. Used for synchronizing the customer file. Through this object customers can be added, modified or deleted.

YSUPPLIER

Supplier object. Used for synchronizing the supplier file. Through this object suppliers can be added, modified or deleted.

YGENACCOUNT

General account object. Used for synchronizing the general account file. Through this object general accounts can be added, modified or deleted.

YFINACCOUNT

Financial account object. Used for maintaining the financial account file. Through this object financial accounts can be added, modified or deleted.

YANALYTICAL

Analytical account object. Used for maintaining the analytical accounts file. Through this object analytical accounts can be added, modified or deleted.

YBOOK

Journal object. Used for maintaining the book file. Through this object sundry, incoming and outgoing books can be added, modified or deleted.

YBOOKING

Interface to pass operations to Wings. You have the possibility to book the operations completely in detail up to the level of individual VAT sections; or, you can let Wings perform the specific bookings of VAT, general discount and financial discount.

YJOURNALCLOSING

This object enables you to close individual journals or groups of journals.

YINFO

This object enables you to query information from Wings such as the outstanding balance of a customer.

YSQL

This object enables you to launch a general SQL query on Wings. The result set is written to an XML file

YCONNECTION

/WINGSACCOUNTING/SESSION/CONNECTION

PROPERTIES:

	Property Name	Type	COM Type	Length	Description	Examples
1	RootPath(*)	String	BSTR	-	Wings.ini file location. Either this property or PathName has to be filled, but it is recommended to use this property.	'C:\Wings'
2	PathName(*) (obsolete)	String	BSTR	-	Wings data location (obsolete: RootPath is the recommended property to be used)	'C:\Wings\Data'
3	DossierID	String	BSTR	4*	Dossier ID. This must designate an existing Wings dossier ID. Defaults to '0001' (fixed Dossier ID for single dossier systems). May only contain 0-9 and spaces. WAI will pad with zeroes if necessary.	<div> " → '0001'</div> <div>'0002' → '0002'</div> <div>'3' → '0003'</div>
4	IniFileName	String	BSTR	-	Name/path of the WAI.INI file (see below). The default will be 'WAI.INI'. If no path is specified, the INI file will be assumed in the dossier path. If the INI file doesn't exist in the designated directory, this will be regarded as an empty INI file.	'C:\Wings\Data\WAI0001.INI'
5	ClientServer	Boolean	BOOL	-	Determines if the client/server database is being used. Default: False (0)	True

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
6	RecordTimeout	Longint	I4	-	Retry time in milliseconds before WAI will return a locking error for record locks. Default: 250 .	500 (= 0.5 sec)
7	TableTimeout	Longint	I4	-	Retry time in milliseconds before WAI will return a locking error for table locks. Default: 1000 .	2000 (= 2 sec)
8	<i>Connected</i>	Boolean	BOOL	-	Connected to Wings dossier?	
9	<i>ErrorCode</i>	Enum	I4	-	Error code. erNone (0): No error ... Is reset to erNone after every successful send.	erNone 2
10	<i>ErrorString</i>	String	BSTR	-	Error as descriptive text.	
11	<i>ErrorProperty</i>	String	BSTR	-	Name of property that caused the error.	'DossierID'
	* Obligatory property <i>Read-only property</i>			* Check length	Default value	

METHODS:

	<i>Method Name</i>	<i>Result Type</i>	<i>COM Type</i>	<i>Description</i>	<i>Special considerations</i>
1	Connect	Boolean	BOOL	<p>Registers a connection to a Wings dossier. The connected object will be used for all other transactions.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if a connection could be established with the designated Wings dossier.</p> <p>False means an error condition. Check ErrorCode and ErrorString for a description of the error.</p>
2	Disconnect	Boolean	BOOL	<p>Disconnects from a Wings dossier.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	False if already disconnected.
3	SetSynchronization Parameter: String	Boolean	BOOL	<p>Sets a new synchronization string that will be remembered by Wings. When the calling application also stores this string, it can be checked with CheckSynchronization in the next WAI session.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>The purpose of these methods is to provide a safety check against a user restoring data of one application and not the other.</p> <p>The calling application has to decide on what to pass. An example could be the date and time of the last connection as a string.</p> <p>The string may be up to 20 characters.</p>

	<i>Method Name</i>	<i>Result Type</i>	<i>COM Type</i>	<i>Description</i>	<i>Special considerations</i>
4	CheckSynchronization Parameter: String	Boolean	BOOL	Checks the last synchronization setting. Result: True (-1): Success False (0): Error	False if the string is not the same as the last one passed with SetSynchronization.

PROPERTIES:

	Property Name	Type	COM Type	Length	Description	Examples
1	Connection	Object	OBJ	-	<p>Connected YConnection object reference. You must set this before any other property.</p> <p>In order to be able to add dossiers, the YConnection object must be connected via the RootPath property.</p>	
2	Action	Enum	I4	-	<p>acCreate (1): creates a new dossier</p> <p>acModify (2): modifies existing dossier data</p> <p>For modification, the current connection is used to identify the dossier. Only general data will be changed (items 7-16).</p>	<p>acCreate</p> <p>1</p>
3	CopyBase	Boolean	BOOL	-	<p>Copy accounting base files from current dossier.</p> <p>Default: False (0)</p>	True
4	CopyCust	Boolean	BOOL	-	<p>Copy customer data from current dossier.</p> <p>May only be set to True when CopyBase is True.</p> <p>Default: False (0)</p>	True
5	CopySupp	Boolean	BOOL	-	<p>Copy supplier data from current dossier.</p> <p>May only be set to True when CopyBase is True.</p> <p>Default: False (0)</p>	True

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
6	PathName*	String	BSTR	-	Dossier data location. Preferably a relative path.	'Wings'
7	DossierCode	String	BSTR	20	Alphanumeric dossier code. If empty, the company name is copied into this property.	'Wings'
8	Name*	String	BSTR	30	Company name.	'Wings nv'
9	Address	String	BSTR	30	Address.	'Bosweg 6'
10	CountryID	String	BSTR	3*	Post office country code. The code must exist in the Wings country file.	'B' 'IRL'
11	CountryISO2	String	BSTR	2*	Two charactered ISO code. The code must exist in the Wings country file.	'BE'
12	PostCode	String	BSTR	10	Post code.	'2220'
13	PlaceName	String	BSTR	25	Place name of the address. Though it is not an error, it is advisable that it is all in capitals.	'HEIST-OP-DEN-BERG'
14	PhoneNumber	String	BSTR	20	Phone number; no checking on format.	
15	LanguageID*	Enum	I4	-	Dossier language. lgDutch (1) lgFrench (2)	

	Property Name	Type	COM Type	Length	Description	Examples
16	VatLiable	Boolean	BOOL	-	Is this company liable for VAT? Default: False (0)	True
17	CompanyNumber	String	BSTR	20	Company number (or VAT number). WAI will convert the VAT number to the correct format when CountryID designates a country of the EU. Must be filled if VatLiable = True.	(CountryISO2 = 'BE') '100000070' → 'BE 0100.000.070' 'be0100.000.070' → 'BE 0100.000.070' 'BE 100-000-070' → 'BE 0100.000.070'
18	FinYearCode1*	String	BSTR	2*	Financial year code of first active financial year.	'09'
19	FinYearCode2*	String	BSTR	2*	Financial year code of second active financial year. Must be different from FinYearCode1.	'10'
20	CurrencyID1	String	BSTR	3*	Currency of first active financial year. Default: EUR	'EUR'
21	CurrencyID2	String	BSTR	3*	Currency of second active financial year. Default: EUR	'EUR'
22	BeginPeriod1*	String	BSTR	6	Starting period of first active financial year in the format 'YYYYMM'.	'200901'
23	BeginPeriod2	String	BSTR	6	Starting period of second active financial year in the format 'YYYYMM'. If empty, it is derived from BeginPeriod1 + 12 months.	'201001'

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
24	BeginPeriod3	String	BSTR	6	Starting period of financial year after the second active financial year in the format 'YYYYMM'. If empty, it is derived from BeginPeriod2 + 12 months.	'201101'
25	<i>NewDossierIDNum</i>	Longint	I4	-	Dossier ID assigned by Wings as an integer value.	201
26	<i>NewDossierIDStr</i>	String	BSTR	4	Dossier ID assigned by Wings as a string.	'0201'
27	<i>ErrorCode</i>	Enum	I4	-	Error code. erNone (0): No error ... Is reset to erNone after every successful send.	erNone 2
28	<i>ErrorString</i>	String	BSTR	-	Error as descriptive text.	
29	<i>ErrorProperty</i>	String	BSTR	-	Name of property that caused the error.	'PathName'

METHODS:

	<i>Method Name</i>	<i>Result Type</i>	<i>COM Type</i>	<i>Description</i>	<i>Special considerations</i>
1	Send	Boolean	BOOL	<p>Sends the transaction to WAI.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if the record could correctly be created or modified.</p> <p>False means an error condition. Check <code>ErrorCode</code> and <code>ErrorString</code> for a description of the error.</p>
2	Validate	Boolean	BOOL	<p>Validates the properties who were set.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if all properties are correct.</p> <p>False means an error condition. Check <code>ErrorCode</code> and <code>ErrorString</code> for a description of the error.</p>
3	Clear	-	-	<p>Resets the object to default values. Also resets <i>ErrorCode</i> and <i>ErrorString</i>.</p>	<p>Use this method if your application must continue using the object after an error condition.</p>

YCUSTOMER

/WINGSACCOUNTING/SESSION/CUSTOMER

PROPERTIES:

	Property Name	Type	COM Type	Length	Description	Examples
1	Connection	Object	OBJ	-	Connected YConnection object reference. You must set this before any other property.	
2	Action	Enum	I4	-	acUpdate (0): creates a new record if non existing CustomerID, modifies the customer information if existing CustomerID acCreate (1): creates a new record; the CustomerID must be empty or nonexistent acModify (2): modifies the customer information; the CustomerID must exist acDelete (3): deletes or inactivates the customer record; the only required property is an existing CustomerID	acModify 2
3	CustomerID	String	BSTR	8*	Existing or new CustomerID or empty. May only contain 0-9 and spaces. WAI will pad with zeroes if necessary.	" → new ID '123' → '00000123' '50000123' → '50000123'

	Property Name	Type	COM Type	Length	Description	Examples
4	AlphaCode	String	BSTR	8	Alphanumeric search code or empty. Though it is not an error, it is advisable that it only contains A-Z and 0-9. WAI will convert to uppercase if necessary.	<div> " → 'WINGSNV' (generated from Name1) </div> <div> 'Wingsnv' → 'WINGSNV' </div>
5	Name1*	String	BSTR	30	Customer name.	'Wings nv'
6	Name2	String	BSTR	30	Second part of customer name or addressee.	
7	Address1	String	BSTR	30	Address part 1.	'Bosweg 6'
8	Address2	String	BSTR	30	Address part 2.	
9	CountryID	String	BSTR	3*	Post office country code. The code must exist in the Wings country file.	<div>'B'</div> <div>'IRL'</div>
10	CountryISO2	String	BSTR	2*	Two charactered ISO code. The code must exist in the Wings country file.	'BE'
11	PostCode	String	BSTR	10	Post code.	'2220'
12	PlaceName	String	BSTR	25	Place name of the address. Though it is not an error, it is advisable that it is all in capitals.	'HEIST-OP-DEN-BERG'

	Property Name	Type	COM Type	Length	Description	Examples
13	LanguageID	Enum	I4	-	lgEmpty (0): will default to the dossier language lgDutch (1) lgFrench (2) lgGerman (3) lgEnglish (4)	
14	VatNumberType	Enum	I4	-	vnNone (0): No VAT number vnNormal (1): VAT number; this will automatically be selected when VatNumber is assigned vnUnknown (2): VAT number unknown vnEntNbr (3): Enterprise number	
15	VatCountryID	String	BSTR	3*	Country code for VAT number. Defaults to CountryID.	'B' 'IRL'
16	VatCountryISO2	String	BSTR	2*	Two characted ISO code for VAT number. The code must exist in the Wings country file.	'BE'

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
17	VatNumber	String	BSTR	20	VAT number. WAI will convert the VAT number to the correct format when VatCountryID designates a country of the EU. The VAT number will be ignored if VatNumberType is not vnNormal.	<i>(VatCountryISO2 = 'BE')</i> '100000070' → 'BE 0100.000.070' 'be0100.000.070' → 'BE 0100.000.070' 'BE 100-000-070' → 'BE 0100.000.070'

	Property Name	Type	COM Type	Length	Description	Examples
18	VatRegime	Enum	I4	-	vrNormal (1): Normal VAT regime for Belgium (automatically selected when the CountryID equals the country of the dossier company) vrEC (2): Normal VAT regime for other EC countries (automatically selected when Country is another EC country) vrImpExp (3): Normal VAT regime for all other countries (automatically selected when Country is none of the above) vrExemption (4): Customer is exempt of VAT vrRegime5 (5): User defined regime vrRegime6 (6): User defined regime vrRegime7 (7): User defined regime vrRegime8 (8): User defined regime	
19	ShowDetail	Boolean	BOOL	-	True (-1): Show account detail False (0): Don't show account detail	

	Property Name	Type	COM Type	Length	Description	Examples
20	CentralizerID*	String	BSTR	8*	General Account number of customer centralizer account. This must be an unused account or an account already used as a customer centralizer account in Wings. This is an obligatory property. A default can be placed in the WAI.INI file.	'400000' '400000 '
21	CurrencyID	String	BSTR	3*	Default currency for this customer. The default currency file of Wings uses ISO codes.	'BEF' 'EUR'
22	BankCountryID	String	BSTR	3*	Country code for bank account number. Defaults to CountryID.	'B' 'IRL'
23	BankCountryISO2	String	BSTR	2*	Two charactered ISO code for bank account number. The code must exist in the Wings country file.	'BE'
24	BankAccount	String	BSTR	34	Bank account number in IBAN format. WAI will convert the Belgian domestic bank number to the IBAN format and automatically adds the correct BicCode.	(BankCountryISO2 = 'BE') '0000000000101' → 'BE43 0000 0000 0101' '000-0000001-01' → 'BE43 0000 0000 0101' 'BE430000000000101' → 'BE43 0000 0000 0101'
25	BicCode	String	BSTR	11	Bank Identifier Code (BIC) also known as SWIFT-code.	'BPOTBEB1'

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
26	Domiciliation	String	BSTR	14	Belgian domiciliation number.	'500123456724' → '500-1234567-24'
27	GenAccountID	String	BSTR	8*	General Account number of default account to be proposed when manually creating an outgoing operation in Wings.	'704000' '704000 '
28	TypeBookingID	String	BSTR	8*	Type Booking number of type booking to be proposed when manually creating an outgoing operation in Wings.	Not supported in Wings standard
29	PaymentCondID*	String	BSTR	4*	Code of Payment Condition defined in Wings. This is an obligatory property. A default can be placed in the WAI.INI file. WAI will convert to uppercase if necessary.	'F008'
30	CreditLimit	Currency	CY	15,2	Credit limit, expressed in the reference currency of the dossier.	100000
31	CreditRestriction	Real	R8	5,2	Credit restriction percentage.	5.8
32	CreditRestrDays	Longint	I4	-	Number of days before applying credit restriction.	8
33	FinDiscount	Real	R8	5,2	Default financial discount percentage.	5
34	FinDiscDays	Longint	I4	-	Number of days that financial discount will be applied.	8
35	GenDiscount	Real	R8	5,2	Default general discount percentage.	2.5

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
36	GroupID*	String	BSTR	4*	Code of Customer group defined in Wings. This is an obligatory property. A default can be placed in the WAI.INI file. WAI will convert to uppercase if necessary.	'USER' '01'
37	RepresentativeID	String	BSTR	4*	Code of default Representative defined in Wings. May be empty. WAI will convert to uppercase if necessary.	'JTR'
38	PaymentComment	String	BSTR	60	Free payment comment.	
39	PhoneNumber	String	BSTR	20	Phone number; no checking on format.	
40	FaxNumber	String	BSTR	20	Fax number; no checking on format.	
41	GsmNumber	String	BSTR	20	GSM number; no checking on format.	
42	Email	String	BSTR	40	E-mail address; no checking on format.	
43	EmailReminder	String	BSTR	40	E-mail address for sending reminder letters.	
44	Memo	String	BSTR	no limit	General comment with no text limit. You can separate different lines with a linefeed.	'General manager: John Doe\n541-555745'
45	UserDefinedField Parameter: String	Variant	VARIANT	-	Get or set a user defined field value. The parameters is the field name.	UserDefinedField['CONTACT']='Johnson'

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
46	<i>NewCustomerIDNum</i>	Longint	I4	-	Customer ID assigned by Wings as an integer value. This is a separate property in order to be able to reset CustomerID to an empty value after a send.	201
47	<i>NewCustomerIDStr</i>	String	BSTR	8	Customer ID assigned by Wings as a string.	'00000201'
48	<i>ErrorCode</i>	Enum	I4	-	Error code. erNone (0): No error ... Is reset to erNone after every successful send.	erNone 2
49	<i>ErrorMessage</i>	String	BSTR	-	Error as descriptive text.	'May not be empty'
50	<i>ErrorProperty</i>	String	BSTR	-	Name of property that caused the error.	'GroupID'

METHODS:

	<i>Method Name</i>	<i>Result Type</i>	<i>COM Type</i>	<i>Description</i>	<i>Special considerations</i>
1	Send	Boolean	BOOL	<p>Sends the transaction to WAI.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if the record could correctly be created/modified/deleted.</p> <p>False means an error condition. Check ErrorCode and ErrorString for a description of the error.</p>
2	Validate	Boolean	BOOL	<p>Validates the properties who were set.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if all properties are correct.</p> <p>False means an error condition. Check ErrorCode and ErrorString for a description of the error.</p>
3	Clear	-	-	<p>Resets the object to default values. Also resets <i>ErrorCode</i> and <i>ErrorString</i>.</p>	<p>Use this method if your application must continue using the object after an error condition.</p>

YSUPPLIER

/WINGSAccounting/Session/Supplier

PROPERTIES:

	Property Name	Type	COM Type	Length	Description	Examples
1	Connection	Object	OBJ	-	Connected YConnection object reference. You must set this before any other property.	
2	Action	Enum	I4	-	acUpdate (0): creates a new record if non existing SupplierID, modifies the supplier information if existing SupplierID acCreate (1): creates a new record; the SupplierID must be empty or nonexistent acModify (2): modifies the supplier information; the SupplierID must exist acDelete (3): deletes or inactivates the supplier record; the only required property is an existing SupplierID	acModify 2
3	SupplierID	String	BSTR	8*	Existing or new SupplierID or empty. May only contain 0-9 and spaces. WAI will pad with zeroes if necessary.	"" → new ID '123' → '00000123' '90000123' → '90000123'

	Property Name	Type	COM Type	Length	Description	Examples
4	AlphaCode	String	BSTR	8	Alphanumeric search code or empty. Though it is not an error, it is advisable that it only contains A-Z and 0-9. WAI will convert to uppercase if necessary.	<div> " → 'WINGSNV' (generated from Name1) </div> <div> 'Wingsnv' → 'WINGSNV' </div>
5	Name1*	String	BSTR	30	Supplier name.	'Wings nv'
6	Name2	String	BSTR	30	Second part of supplier name or addressee.	
7	Address1	String	BSTR	30	Address part 1.	'Bosweg 6'
8	Address2	String	BSTR	30	Address part 2.	
9	CountryID	String	BSTR	3*	Post office country code. The code must exist in the Wings country file.	<div>'B'</div> <div>'IRL'</div>
10	CountryISO2	String	BSTR	2*	Two charactered ISO code. The code must exist in the Wings country file.	'BE'
11	PostCode	String	BSTR	10	Post code.	'2220'
12	PlaceName	String	BSTR	25	Place name of the address. Though it is not an error, it is advisable that it is all in capitals.	'HEIST-OP-DEN-BERG'

	Property Name	Type	COM Type	Length	Description	Examples
13	LanguageID	Enum	I4	-	lgEmpty (0): will default to the dossier language lgDutch (1) lgFrench (2) lgGerman (3) lgEnglish (4)	
14	VatNumberType	Enum	I4	-	vnNone (0): No VAT number vnNormal (1): VAT number; this will automatically be selected when VatNumber is assigned vnUnknown (2): VAT number unknown vnEntNbr (3): Enterprise number	
15	VatCountryID	String	BSTR	3*	Country code for VAT number. Defaults to CountryID.	'B' 'IRL'
16	VatCountryISO2	String	BSTR	2*	Two characted ISO code for VAT number. The code must exist in the Wings country file.	'BE'

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
17	VatNumber	String	BSTR	20	VAT number. WAI will convert the VAT number to the correct format when CountryID designates a country of the EU. The VAT number will be ignored if VatNumberType is not vnNormal.	<i>(CountryID = 'B')</i> '000000097' → 'BE 000.000.097' 'be000.000.097' → 'BE 000.000.097' 'BE 000-000-097' → 'BE 000.-00.0-097'

	Property Name	Type	COM Type	Length	Description	Examples
18	VatRegime	Enum	I4	-	vrNormal (1): Normal VAT regime for Belgium (automatically selected when the CountryID equals the country of the dossier company) vrEC (2): Normal VAT regime for other EC countries (automatically selected when Country is another EC country) vrImpExp (3): Normal VAT regime for all other countries (automatically selected when Country is none of the above) vrExemption (4): Supplier is exempt of VAT vrRegime5 (5): User defined regime vrRegime6 (6): User defined regime vrRegime7 (7): User defined regime vrRegime8 (8): User defined regime	
19	ShowDetail	Boolean	BOOL	-	True (-1): Show account detail False (0): Don't show account detail	

	Property Name	Type	COM Type	Length	Description	Examples
20	CentralizerID*	String	BSTR	8*	General Account number of supplier centralizer account. This must be an unused account or an account already used as a supplier centralizer account in Wings. This is an obligatory property. A default can be placed in the WAI.INI file.	'440000' '440000 '
21	CurrencyID	String	BSTR	3*	Default currency for this supplier. The default currency file of Wings uses ISO codes.	'BEF' 'EUR'
22	BankCountryID	String	BSTR	3*	Country code for bank account number. Defaults to CountryID.	'B' 'IRL'
23	BankCountryISO2	String	BSTR	2*	Two charactered ISO code for bank account number. The code must exist in the Wings country file.	'BE'
24	BankAccount	String	BSTR	34	Bank account number in IBAN format. WAI will convert the Belgian domestic bank number to the IBAN format and automatically adds the correct BicCode.	(BankCountryISO2 = 'BE') '0000000000101' → 'BE43 0000 0000 0101' '000-0000001-01' → 'BE43 0000 0000 0101' 'BE430000000000101' → 'BE43 0000 0000 0101'
25	BicCode	String	BSTR	11	Bank Identifier Code (BIC) also known as SWIFT-code.	'BPOTBEB1'

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
26	Domiciliation	String	BSTR	14	Belgian domiciliation number.	'500123456724' → '500-1234567-24'
27	GenAccountID	String	BSTR	8*	General Account number of default account to be proposed when manually creating an outgoing operation in Wings.	'604000' '604000 '
28	TypeBookingID	String	BSTR	8*	Type Booking number of type booking to be proposed when manually creating an outgoing operation in Wings.	Not supported in Wings standard
29	PaymentCondID*	String	BSTR	4*	Code of Payment Condition defined in Wings. This is an obligatory property. A default can be placed in the WAI.INI file. WAI will convert to uppercase if necessary.	'M030'
30	CreditLimit	Currency	CY	15,2	Credit limit, expressed in the reference currency of the dossier.	100000
31	CreditRestriction	Real	R8	5,2	Credit restriction percentage.	5.8
32	CreditRestrDays	Longint	I4	-	Number of days before applying credit restriction.	8
33	FinDiscount	Real	R8	5,2	Default financial discount percentage.	5
34	FinDiscDays	Longint	I4	-	Number of days that financial discount will be applied.	8
35	GenDiscount	Real	R8	5,2	Default general discount percentage.	2.5

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
36	GroupID*	String	BSTR	4*	Code of Supplier group defined in Wings. This is an obligatory property. A default can be placed in the WAI.INI file. WAI will convert to uppercase if necessary.	'FABR' '01'
37	AutomaticPayment	Boolean	BOOL	-	True: Pay with Wings Banking Module False	
38	PaymentComment	String	BSTR	60	Free payment comment.	
39	PhoneNumber	String	BSTR	20	Phone number; no checking on format.	
40	FaxNumber	String	BSTR	20	Fax number; no checking on format.	
41	GsmNumber	String	BSTR	20	GSM number; no checking on format.	
42	Email	String	BSTR	40	E-mail address; no checking on format.	
43	Memo	String	BSTR	no limit	General comment with no text limit. You can separate different lines with a linefeed.	'General manager: John Doe\n541-555745'
44	UserDefinedField Parameter: String	Variant	VARIANT	-	Get or set a user defined field value. The parameters is the field name.	UserDefinedField['CONTACT']='Johnson'
45	NewSupplierIDNum	Longint	I4	-	Supplier ID assigned by Wings as an integer value. This is a separate property in order to be able to reset SupplierID to an empty value after a send.	201

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
46	<i>NewSupplierIDStr</i>	String	BSTR	8	Supplier ID assigned by Wings as a string.	'00000201'
47	<i>ErrorCode</i>	Enum	I4	-	<p>Error code.</p> <p>erNone (0): No error</p> <p>...</p> <p>Is reset to erNone after every successful send.</p>	<p>erNone</p> <p>2</p>
48	<i>ErrorString</i>	String	BSTR	-	Error as descriptive text.	'May not be empty'
49	<i>ErrorProperty</i>	String	BSTR	-	Name of property that caused the error.	'GroupID'

METHODS:

	<i>Method Name</i>	<i>Result Type</i>	<i>COM Type</i>	<i>Description</i>	<i>Special considerations</i>
1	Send	Boolean	BOOL	<p>Sends the transaction to WAI.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if the record could correctly be created/modified/deleted.</p> <p>False means an error condition. Check <code>ErrorCode</code> and <code>ErrorString</code> for a description of the error.</p>
2	Validate	Boolean	BOOL	<p>Validates the properties who were set.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if all properties are correct.</p> <p>False means an error condition. Check <code>ErrorCode</code> and <code>ErrorString</code> for a description of the error.</p>
3	Clear	-	-	<p>Resets the object to default values. Also resets <i>ErrorCode</i> and <i>ErrorString</i>.</p>	<p>Use this method if your application must continue using the object after an error condition.</p>

YGenAccount

/WINGSACCOUNTING/SESSION/GENACCOUNT

PROPERTIES:

	Property Name	Type	COM Type	Length	Description	Examples
1	Connection	Object	OBJ	-	Connected YConnection object reference. You must set this before any other property.	
2	Action	Enum	I4	-	acUpdate (0): creates a new record if non existing AccountNumber, modifies the general account information if existing AccountNumber acCreate (1): creates a new record; the AccountNumber may not already exist acModify (2): modifies the general account information; the AccountNumber must exist acDelete (3): deletes or inactivates the general account record; the only required property is an existing AccountNumber	acModify 2
3	AccountNumber*	String	BSTR	8*	Existing or new account number. May only contain 0-9 and spaces.	'700000' → '700000' '70000000' → '70000000'

	Property Name	Type	COM Type	Length	Description	Examples
4	AlphaCode	String	BSTR	8	Alphanumeric search code or empty. Though it is not an error, it is advisable that it only contains A-Z and 0-9. WAI will convert to uppercase if necessary.	" → 'KAPITAAL' (generated from Name01) 'Kapitaal' → 'KAPITAAL'
5	Name01*	String	BSTR	30	General account name language 01.	'Kapitaal'
6	Name02	String	BSTR	30	General account name language 02.	'Capital'
7	Name03	String	BSTR	30	General account name language 03.	'Capital'
8	Name04	String	BSTR	30	General account name language 04.	'Kapital'
9	ShowDetail	Boolean	BOOL	-	True (-1): Show gen. account detail False (0): Don't show gen.account detail	-1
10	Settle	Boolean	BOOL	-	True (-1): You can immediately settle a financial/sundry operation. False (0): Settlement later	-1
11	CurrencyID	String	BSTR	3*	Default currency for this general account. The default currency file of Wings uses ISO codes.	'EUR'
12	HasAnalytical	Boolean	BOOL	-	True (-1): You can use analytical accounts False (0): You don't use analytical accounts	

	Property Name	Type	COM Type	Length	Description	Examples
13	AnalyticalAccountID (obsolete)	String	BSTR	8*	Analytical account. (obsolete: this property is identical to AnalyticalAccount1 and is only kept for backward compatibility)	
14	AnalyticalAccount1	String	BSTR	8*	If HasAnalytical is True you can set here your analytical account for the 1 st distribution. The account must exist in Wings.	'90012'
15	AnalyticalAccount2	String	BSTR	8*	Idem for the 2 nd analytical distribution.	
16	AnalyticalAccount3	String	BSTR	8*	Idem for the 3 rd analytical distribution.	
17	AnalyticalAccount4	String	BSTR	8*	Idem for the 4 th analytical distribution.	
18	AnalyticalAccount5	String	BSTR	8*	Idem for the 5 th analytical distribution.	
19	HasQuantities	Boolean	BOOL	-	<p>True (-1): Keep quantities on this account</p> <p>False (0): No quantities</p>	
20	Units	String	BSTR	3*	Unit code (not checked).	'Kg'
21	QuantityDecimals	Longint	I4	1	<p>Number of decimals for quantities.</p> <p>Default : 1 Min. : 0 Max. : 4</p>	1
22	Memo	Boolean	BOOL	-	General comment with not text limit. You Can separate different lines with a linefeed.	
23	UserDefinedField Parameter: String	Variant	VARIANT	-	Get or set a user defined field value. The parameters is the field name.	UserDefinedField['HLDACC']='AC85004'

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
24	<i>ErrorCode</i>	Enum	I4	-	Error code. erNone (0): No error ... Is reset to erNone after every successful send.	ErNone 2
25	<i>ErrorString</i>	String	BSTR	-	Error as descriptive text.	'May not be empty'
26	<i>ErrorProperty</i>	String	BSTR	-	Name of property that caused the error.	'Name01'

METHODS:

	<i>Method Name</i>	<i>Result Type</i>	<i>COM Type</i>	<i>Description</i>	<i>Special considerations</i>
1	Send	Boolean	BOOL	<p>Sends the transaction to WAI.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if the record could correctly be created/modified/deleted.</p> <p>False means an error condition. Check <code>ErrorCode</code> and <code>ErrorString</code> for a description of the error.</p>
2	Validate	Boolean	BOOL	<p>Validates the properties who were set.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if all properties are correct.</p> <p>False means an error condition. Check <code>ErrorCode</code> and <code>ErrorString</code> for a description of the error.</p>
3	Clear	-	-	<p>Resets the object to default values. Also resets <i>ErrorCode</i> and <i>ErrorString</i>.</p>	<p>Use this method if your application must continue using the object after an error condition.</p>

PROPERTIES:

	Property Name	Type	COM Type	Length	Description	Examples
1	Connection	Object	OBJ	-	Connected YConnection object reference. You must set this before any other property.	
2	Action	Enum	I4	-	acUpdate (0): creates a new record if non existing AccountNumber, modifies the analytical account information if existing AccountNumber acCreate (1): creates a new record; the AccountNumber may not already exist acModify (2): modifies the account information; the AccountNumber must exist acDelete (3): deletes or inactivates the account record; the only required property is an existing AccountNumber	acModify 2
3	AccountNumber*	String	BSTR	8*	Existing or new account number. May only contain 0-9 and spaces.	'800123'

	Property Name	Type	COM Type	Length	Description	Examples
4	AlphaCode	String	BSTR	8	Alphanumeric search code or empty. Though it is not an error, it is advisable that it only contains A-Z and 0-9. WAI will convert to uppercase if necessary.	" → 'DEPTARTM' (generated from Name01) 'DEPT1' → 'DEPT1'
5	Name01*	String	BSTR	30	Account description language 01. Obligatory with action = acCreate	
6	Name02	String	BSTR	30	Account description language 02.	
7	Name03	String	BSTR	30	Account description language 03.	
8	Name04	String	BSTR	30	Account description language 04.	
9	Sort (obsolete)	Enum	I4	-	asNormal (0): normal analytical account (equivalent: Distribution=1) asProject (1): project account (equivalent: Distribution=2)	asProject 1
10	Distribution	Longint	I4	1	Analytical distribution 1-5. Default is 1.	2
11	TransferAmount	Currency	CY	15,2	Transfer amount for beginning of first financial year (only for project accounts).	14500.50
12	TransferQuantity	Real	R8	14,4	Transfer amount for beginning of first financial year (only for project accounts with quantities).	250.0000

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
13	HasDistribution	Boolean	BOOL	-	<p>True (-1): Account is a distribution account (always manual distribution)</p> <p>False (0): 'Normal' detail account</p>	
14	ShowDetail	Boolean	BOOL	-	<p>True (-1): Show account detail</p> <p>False (0): Don't show account detail</p>	
15	HasQuantities	Boolean	BOOL	-	<p>True (-1): Quantities are to be registered for this account</p> <p>False (0): No quantities</p>	
16	Units	String	BSTR	3*	Unit code (not checked).	'kg'
17	QuantityDecimals	Longint	I4	1	<p>Number of decimals for quantities.</p> <p>Default : 1 Min. : 0 Max. : 4</p>	1
18	Memo	String	BSTR	no limit	General comment with no text limit. You can separate different lines with a linefeed.	
19	UserDefinedField Parameter: String	Variant	VARIANT	-	Get or set a user defined field value. The parameters is the field name.	UserDefinedField['HLDACC']='AC85004'

METHODS:

	<i>Method Name</i>	<i>Result Type</i>	<i>COM Type</i>	<i>Description</i>	<i>Special considerations</i>
1	Send	Boolean	BOOL	<p>Sends the transaction to WAI.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if the record could correctly be created/modified/deleted.</p> <p>False means an error condition. Check <code>ErrorCode</code> and <code>ErrorString</code> for a description of the error.</p>
2	Validate	Boolean	BOOL	<p>Validates the properties who were set.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if all properties are correct.</p> <p>False means an error condition. Check <code>ErrorCode</code> and <code>ErrorString</code> for a description of the error.</p>
3	Clear	-	-	<p>Resets the object to default values. Also resets <i>ErrorCode</i> and <i>ErrorString</i>.</p>	<p>Use this method if your application must continue using the object after an error condition.</p>

PROPERTIES:

	Property Name	Type	COM Type	Length	Description	Examples
1	Connection	Object	OBJ	-	Connected YConnection object reference. You must set this before any other property.	
2	Action	Enum	I4	-	<p>acUpdate (0): creates a new record if non existing AccountNumber, modifies the financial account information if existing AccountNumber</p> <p>acCreate (1): creates a new record; the AccountNumber may not already exist</p> <p>acModify (2): modifies the financial account information; the AccountNumber must exist</p> <p>acDelete (3): deletes or inactivates the financial account record; the only required property is an existing AccountNumber</p>	<p>acModify</p> <p>2</p>
3	BookCode*	String	BSTR	4*	<p>Code. Though it is not an error, it is advisable that it only contains A-Z and 0-9.</p> <p>WAI will convert to uppcase if necessary.</p>	'BNK2'

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
4	AccountNumber*	String	BSTR	8*	General account number. May only contain 0-9 and spaces.	'550010'
5	Name01*	String	BSTR	30	Book name language 01.	'Bank 2'
6	Name02	String	BSTR	30	Book name language 02.	'Banque 2'
7	Name03	String	BSTR	30	Book name language 03.	'Bank 2'
8	Name04	String	BSTR	30	Book name language 04.	'Bank 2'
9	NumberingType*	Enum	I4	-	ntPerpetual (1) Perpetual numbering ntCalYear (2) Numbering per calendar year ntFinYear (3) Numbering per financial year	ntFinYear 3
10	ImportProtection	Boolean	BOOL	-	True (-1): Makes the book only usable via this interface. False (0): You can also use this book directly in Wings.	
11	CurrencyID	String	BSTR	3*	Currency for this financial book. The default currency file of Wings uses ISO codes.	'EUR'
12	NegAccountID	String	BSTR	8*	General Account number for booking of negative balance while closing the financial year.	'430010'

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
13	BankCountryID	String	BSTR	3*	Country code for bank account number. Defaults to dossier country.	'B' 'IRL'
14	BankCountryISO2	String	BSTR	2*	Two characted ISO code for bank account number. The code must exist in the Wings country file.	'BE'
15	BankAccount	String	BSTR	34	Bank account number in IBAN format. WAI will convert the Belgian domestic bank number to the IBAN format and automatically adds the correct BicCode.	(BankCountryISO2 = 'BE') '000000000101' → 'BE43 0000 0000 0101' '000-0000001-01' → 'BE43 0000 0000 0101' 'BE43000000000101' → 'BE43 0000 0000 0101'
16	BicCode	String	BSTR	11	Bank Identifier Code (BIC) also known as SWIFT-code.	'BPOTBEB1'
17	Memo	String	BSTR	no limit	General comment with no text limit. You can separate different lines with a linefeed.	
18	UserDefinedField Parameter: String	Variant	VARIANT	-	Get or set a user defined field value. The parameters is the field name.	UserDefinedField['HLDACC']='AC85004'

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
19	<i>ErrorCode</i>	Enum	I4	-	Error code. erNone (0): No error ... Is reset to erNone after every successful send.	ErNone 2
20	<i>ErrorString</i>	String	BSTR	-	Error as descriptive text.	'May not be empty'
21	<i>ErrorProperty</i>	String	BSTR	-	Name of property that caused the error.	'Name01'

METHODS:

	<i>Method Name</i>	<i>Result Type</i>	<i>COM Type</i>	<i>Description</i>	<i>Special considerations</i>
1	Send	Boolean	BOOL	<p>Sends the transaction to WAI.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if the record could correctly be created/modified/deleted.</p> <p>False means an error condition. Check ErrorCode and ErrorString for a description of the error.</p>
2	Validate	Boolean	BOOL	<p>Validates the properties who were set.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if all properties are correct.</p> <p>False means an error condition. Check ErrorCode and ErrorString for a description of the error.</p>
3	Clear	-	-	<p>Resets the object to default values. Also resets <i>ErrorCode</i> and <i>ErrorString</i>.</p>	<p>Use this method if your application must continue using the object after an error condition.</p>

PROPERTIES:

	Property Name	Type	COM Type	Length	Description	Examples
1	Connection	Object	OBJ	-	Connected YConnection object reference. You must set this before any other property.	
2	Action	Enum	I4	-	acUpdate (0): creates a new record if non existing BookCode, modifies the book information if existing BookCode acCreate (1): creates a new record; the BookCode may not already exist acModify (2): modifies the book information; the BookCode must exist acDelete (3): deletes or inactivates the book record; the only required property is an existing BookCode	acModify 2
3	BookCode*	String	BSTR	4*	Code. Though it is not an error, it is advisable that it only contains A-Z and 0-9. WAI will convert to uppercase if necessary.	'V02'
4	Name01*	String	BSTR	30	Book name language 01.	'Verkopen 2'

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>		<i>Examples</i>
5	Name02	String	BSTR	30	Book name language 02.		'Ventes 2'
6	Name03	String	BSTR	30	Book name language 03.		'Verkäufe 2'
7	Name04	String	BSTR	30	Book name language 04.		'Sales 2'
8	BookType*	Enum	I4	-	btIncoming (1) btOutgoing (2) btSundry (3) btClosing (4) btSimulation (6) btContingent (7) btAnalytical (8)		btOutgoing 2
9	NumberingType*	Enum	I4	-	ntPerpetual (1) Perpetual numbering ntCalYear (2) Numbering per calendar year ntFinYear (3) Numbering per financial year		ntFinYear 3
10	ImportProtection	Boolean	BOOL	-	True (-1): Makes the book only usable via this interface False (0): You can also use this book directly in Wings		

	Property Name	Type	COM Type	Length	Description	Examples
11	ProForma	Boolean	BOOL		<p>True (-1): Special pro forma book. Only to be used for incoming our outgoing operations</p> <p>False (0): Regular book</p>	
12	ErrorCode	Enum	I4	-	<p>Error code.</p> <p>erNone (0): No error</p> <p>...</p> <p>Is reset to erNone after every successful send.</p>	<p>ErNone</p> <p>2</p>
13	ErrorString	String	BSTR	-	Error as descriptive text.	'May not be empty'
14	ErrorProperty	String	BSTR	-	Name of property that caused the error.	'Name01'

METHODS:

	<i>Method Name</i>	<i>Result Type</i>	<i>COM Type</i>	<i>Description</i>	<i>Special considerations</i>
1	Send	Boolean	BOOL	<p>Sends the transaction to WAI.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if the record could correctly be created/modified/deleted.</p> <p>False means an error condition. Check <code>ErrorCode</code> and <code>ErrorString</code> for a description of the error.</p>
2	Validate	Boolean	BOOL	<p>Validates the properties who were set.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if all properties are correct.</p> <p>False means an error condition. Check <code>ErrorCode</code> and <code>ErrorString</code> for a description of the error.</p>
3	Clear	-	-	<p>Resets the object to default values. Also resets <i>ErrorCode</i> and <i>ErrorString</i>.</p>	<p>Use this method if your application must continue using the object after an error condition.</p>

YBOOKING

/WINGSACCOUNTING/SESSION/BOOKINGBATCH/BOOKING

PROPERTIES:

	Property Name	Type	COM Type	Length	Description	Examples
Booking Header						/WINGSACCOUNTING/SESSION/BOOKINGBATCH/BOOKING/HEADER
1	Connection	Object	OBJ	-	Connected YConnection object reference. You should set this property only once, and before any other property is assigned.	
2	Action	Enum	I4	-	acCreate (1): creates a new booking acModify (2): modifies an existing booking BookCode, RunningNumber and Period must designate an existing booking.	acCreate 1
3	BookCode*	String	BSTR	4*	Existing book code. The code must exist in the Wings book file.	'V01'

	Property Name	Type	COM Type	Length	Description	Examples
4	RunningNumber	String	BSTR	8*	Existing or new running number or empty. May only contain 0-9 and spaces. WAI will pad with zeroes if necessary. If the given number contains 6 digits or less, and the book's numbering type is per calendar or financial year, Wings will add a prefix according to its numbering rules.	'98000201' → '98000201' " → '98000201' <i>(numbering per calendar year)</i> " → '98000201' <i>(assigned by Wings)</i>
5	Period	String	BSTR	6	Period in the format 'YYYYMM'. This property is ignored for financial operations. If empty, it is derived from OpDate. The period must be in a valid range according to closed periods and open financial years.	'19980923' → '199809' " → '199809' <i>(derived from OpDate)</i>
6	OpDate*	String	BSTR	8*	Date of the operation in the format 'YYYYMMDD' (named OpDate instead of Date to avoid reserved word conflicts). OpDate is the invoice date for incoming and outgoing invoices.	'19980923'

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
7	VatDate	String	BSTR	8*	<p>Date of VAT collectability in the format 'YYYYMMDD'.</p> <p>Only to be used for incoming and outgoing invoices (not pro forma).</p> <p>Defaults to OpDate.</p>	
8	RepresentativeID	String	BSTR	4*	<p>Representative code to which the turnover will be assigned to, defined in Wings. Ignored for all operations but outgoing.</p> <p>May be empty.</p> <p>WAI will convert to uppercase if necessary.</p>	'JTR'
9	PaymentCondID	String	BSTR	4*	<p>Code of Payment Condition defined in Wings or empty.</p> <p>This property is only used to calculate the due dates if necessary. It is also shown when modifying the operation in Wings.</p> <p>If empty it defaults to the customer/supplier value for incoming or outgoing operations. It is ignored for other operations.</p> <p>WAI will convert to uppercase if necessary.</p>	'F008'

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
10	FinDiscount	Real	R8	5,2	Informative financial discount percentage. This property will only be used to calculate a default financial discount amount when settling an account in Wings. Shown when modifying the operation in Wings.	5
11	FinDiscExempt	Boolean	BOOL		True (-1): Financial discount is exempt of VAT (default value for Belgium) False (0)	
12	GenDiscount	Real	R8	5,2	Purely informative general discount percentage; this property will not be used for calculation. Shown when modifying the operation in Wings.	2.5
13	HdrCurrencyID	String	BSTR	3*	Currency code of the operation. If empty, the base currency of the financial year is assumed. Sundry and financial operations may state a different currency per detail line (see below). The default currency file of Wings uses ISO codes.	'EUR' 'USD'

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
14	HdrExchange	Real	R8	11,6	Exchange rate of the currency expressed in base currency units. Ignored if HdrCurrencyID designates a currency with a fixed exchange rate relative to the base currency.	38.574
15	HdrIndExchange	Real	R8	11,6	Indirect exchange rate of the currency expressed in base currency units. Ignored if HdrCurrencyID designates a currency with a fixed exchange rate relative to the base currency.	0.025924
16	DocumentSort(*)	Enum	I4	-	<p>dsNone (0): Not applicable</p> <p>You need to specify one of the following for incoming and outgoing operations:</p> <p>dsInvoice (1): Invoice</p> <p>dsCreditNote (2): Credit note</p> <p>You need to specify one of the following if you are using a closing book with a starting period:</p> <p>dsOBGeneral (3): Opening balance general accounts</p> <p>dsOBCustomer (4): Opening balance customers</p> <p>dsOBSupplier (5): Opening balance suppliers</p> <p>dsOBFinancial (6): Opening balance financial accounts</p>	<p>dsInvoice</p> <p>1</p>

	Property Name	Type	COM Type	Length	Description	Examples
17	SignerID	String	BSTR	4*	Invoice signer code. Ignored for all operations but incoming. May be empty.	'JTR'
18	Definitive	Enum	I4	-	dfUnknown (0): will default to the current setting in Wings dfDefinitive (1): definitive booking dfPreliminary (2): preliminary booking dfToBeChecked (3): preliminary and to be checked in Wings You can make a preliminary booking definitive by modifying it, the opposite is not possible.	dfUnknown → dfDefinitive
19	Blocked	Boolean	BOOL	-	Supplier record will be blocked for payment when true. Ignored for all operations but incoming.	True -1
20	Memo	String	BSTR	no limit	Internal booking comment with no text limit. You can separate different lines with a linefeed.	'Account correction'

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
21	<i>NewRunNumberNum</i>	Longint	I4	-	<p>Running number assigned by Wings as an integer value. This is a separate property in order to be able to reset RunningNumber to an empty value after a send.</p> <p>This property can only be returned by WAI after a call to SendBatch (see further) so it only reflects the number of the last booking. This means that if you need feedback on the numbering performed by Wings, you have to call SendBatch after each call to SendBooking.</p>	98000201
22	<i>NewRunNumberStr</i>	String	BSTR	8	Running number assigned by Wings as a string.	'98000201'

Property Name	Type	COM Type	Length	Description	Examples
Booking Detail (not to be used for purely analytical bookings)					/WINGSACCOUNTING/SESSION/BOOKINGBATCH/BOOKING/DETAIL
23	AccountType	Enum	I4	-	<p>atGeneral (1): General account</p> <p>atCustomer (2): Customer account</p> <p>atSupplier (3): Supplier account</p> <p>atProFormaCustomer (6): Pro forma customer</p> <p>atProFormaSupplier (7): Pro forma supplier</p> <p>The first record and that alone of an incoming or outgoing operation must be a supplier/customer record respectively. The first record and that alone of a financial operation must be the financial account linked to the financial book.</p> <p>Pro forma customers/suppliers are only allowed for sundry, closing, financial and pro forma operations. A pro forma operations has only one detail record.</p>

	Property Name	Type	COM Type	Length	Description	Examples
24	AccountID*	String	BSTR	8*	<p>General Account number or Customer or Supplier ID. This must be an existing account.</p> <p>The account may not be a customer or supplier centralizer account and may not be a financial account except for the first record of a financial operation.</p> <p>This is an obligatory property.</p>	<p>(general account)</p> <p>'704000'</p> <p>'704000 '</p> <p>(customer/supplier)</p> <p>'201' → '00000201'</p>
25	Comment	String	BSTR	128	<p>A comment for this booking line.</p> <p>For incoming and outgoing operations, this is the first part of the supplier/customer name respectively if empty.</p> <p>This comment is visible on the officially printed journal.</p>	
26	DetCurrencyID	String	BSTR	3*	<p>Currency ID. Supposed identical to HdrCurrencyID when empty.</p> <p>Must be empty or equal to HdrCurrencyID for incoming and outgoing operations.</p>	<p>''</p> <p>'BEF'</p> <p>'EUR'</p>
27	DetExchange	Real	R8	11,6	<p>Exchange rate of the currency expressed in base currency units. Ignored if DetCurrencyID designates a currency with a fixed exchange rate relative to the base currency. Must be zero or identical to HdrCurrencyID for incoming and outgoing operations.</p>	38.574

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
28	DetIndExchange	Real	R8	11,6	Indirect exchange rate of the currency expressed in base currency units. Ignored if DetCurrencyID designates a currency with a fixed exchange rate relative to the base currency.	0.025924

	Property Name	Type	COM Type	Length	Description	Examples
29	LineAmountDC*	Currency	CY	15,2	<p>Line amount in detail currency (expressed in DetCurrencyID units). If Wings must book several entries for the booking detail object, this amount is the total of all entries for that object.</p> <p><u>General rules for amounts:</u></p> <p>All amounts have a sign according to the following rule:</p> <ul style="list-style-type: none"> • Debit is positive (+) • Credit is negative (-) <p>The number of decimals used in the amount must correspond to the financial unit of the applied currency (e.g. 0.01 for EUR, USD; 1.00 for JPY). Wings will reject the transaction if this is not the case.</p> <p>Amounts may be zero. However, at least one entry will be booked for a booking detail object with all zero amounts.</p> <p>If values in different currencies are assigned, WAI will check that they are within a certain range from their internally calculated counterparts (always with respect to base currency). The maximal deviation is 2*[smallest financial unit] or 1 % for non-euro currencies (if this is bigger).</p>	-1361.25 (EUR) (1 EUR = 40.2956 BEF)

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
30	LineAmountBC	Currency	CY	15,2	<p>Line amount expressed in the base currency.</p> <p>If this property is not assigned, it will be calculated from LineAmountDC.</p> <p>Notice however, that if LineAmountHC (see below) is not used, this property will be checked to see if the booking is in balance. That means that either LineAmountBC or LineAmountHC should always be assigned.</p>	-54853 (BEF) (sum of BaseAmountBC and VatAmountBC)
31	LineAmountHC	Currency	CY	15,2	<p>Line amount expressed in HdrCurrencyID units.</p> <p>Within one booking, you should either always assign LineAmountHC or never.</p> <p>If assigned, this property will be used to check if the booking is in balance. A rounding difference entry will then eventually be booked automatically by Wings to balance the total amount in the base currency.</p> <p>Pro forma operations are automatically balanced because the counter detail record is generated as the mirrored amount of the one detail record.</p>	-1361.25 (EUR) (= LineAmountDC)

	Property Name	Type	COM Type	Length	Description	Examples
32	DiscAmountDC	Currency	CY	15,2	<p>General discount amount in detail currency.</p> <p>This property should normally have the opposite sign of LineAmountDC. This amount is not separately added to LineAmountDC because it should already be subtracted from the VAT amounts (see examples). Therefore, it is only useful to assign to this property if Wings is setup to book the general discount on a separate account.</p> <p>Not applicable to sundry and financial operations.</p>	125.00 (EUR)
33	DiscAmountBC	Currency	CY	15,2	<p>General discount amount expressed in the base currency.</p> <p>Calculated automatically if unassigned.</p>	5037.00 (BEF)
34	FinDiscAmountDC	Currency	CY	15,2	<p>Base of financial discount. If not assigned, and FinDiscExempt is True, this value is automatically calculated as the difference between LineAmountDC and the sum of BaseAmountDC and VatAmountDC.</p> <p>Not applicable to sundry and financial operations.</p>	

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
35	FinDiscAmountBC	Currency	CY	15,2	Base of financial discount expressed in the base currency. Calculated automatically if unassigned.	
36	BaseAmountDC	Currency	CY	15,2	Base of VAT amount in detail currency. Only used if VatCode is assigned. If used in financial operations (for bank charges) the difference between the line amount and the sum of the other amounts is supposed to be without VAT. If used in incoming and outgoing operations, the difference between the line amount and the sum of the other amounts is supposed to be the base of financial discount. This will eventually be booked on a separate account according to the normal Wings procedure. Not applicable to sundry operations.	-1125.00 (EUR)

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
37	BaseAmountBC	Currency	CY	15,2	<p>Base of VAT amount expressed in the base currency.</p> <p>Calculated automatically if unassigned. It is advisable to assign this property explicitly though, because it directly impacts the VAT Declaration.</p> <p>You should also ensure that rounding differences don't show in the sum of the amounts in the base currency to avoid them being booked as financial discounts (incoming/outgoing operations) or separate entries w/o VAT (financial operations).</p>	-45333.00 (BEF) (-1125.00 * 40.2956)
38	VatAmountDC	Currency	CY	15,2	VAT amount in detail currency. Only used if VatCode is used.	-236.25 (EUR) (21% of BaseAmountDC)
39	VatAmountBC	Currency	CY	15,2	<p>VAT amount expressed in the base currency.</p> <p>Calculated automatically if unassigned. It is advisable to assign this property explicitly though, because it directly impacts the VAT Declaration.</p>	-9520.00 (BEF) (-236.25 * 40.2956)

	Property Name	Type	COM Type	Length	Description	Examples
40	CommercialTurnover	Boolean	BOOL	-	<p>Register base of VAT amount as commercial turnover if True. This will show in Wings in the turnover charts of customers, representatives and suppliers.</p> <p>Only applicable for incoming and outgoing operations when using VAT sections rather than VAT codes.</p> <p>Ignored in all other cases.</p>	<p>True</p> <p>-1</p>
41	VatCodeType	Enum	I4	-	<p>vcNone (0): No VAT code used. Obligatory setting for sundry operations and customer, supplier or financial account entries.</p> <p>vcIncoming (1): VAT code for incoming operations. Not to be used in sundry and outgoing operations.</p> <p>vcOutgoing (2): VAT code for outgoing operations. Not to be used in sundry and incoming operations.</p>	<p>vcIncoming</p> <p>1</p>

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
42	VatCode	String	BSTR	5*	<p>VAT code using VatCodeType. Ignored if VatCodeType equals vcNone.</p> <p>Using VAT codes can greatly simplify the booking of incoming and outgoing operations. This approach is recommended over using VAT sections, which is more low level and demands more knowledge of accounting practices.</p> <p>Check Wings for valid VAT codes.</p>	'21'
43	VatSection1	String	BSTR	5*	<p>First VAT section. Up to three sections can be used simultaneously.</p> <p>Ignored if a VAT code is used.</p> <p>Check Wings for valid VAT section codes.</p>	'03'
44	VatSection2	String	BSTR	5*	Second VAT section.	
45	VatSection3	String	BSTR	5*	Third VAT section.	
46	Quantity	Real	R8	14,4	Quantity for general or analytical account .	5.00
47	Analytical1	String	BSTR	8*	Analytical account number for analytical distribution 1.	'810201'
48	Analytical2	String	BSTR	8*	Analytical account number for analytical distribution 2.	
49	Analytical3	String	BSTR	8*	Analytical account number for analytical distribution 3.	

	Property Name	Type	COM Type	Length	Description	Examples	
50	Analytical4	String	BSTR	8*	Analytical account number for analytical distribution 4.		
51	Analytical5	String	BSTR	8*	Analytical account number for analytical distribution 5.		
52	DueDate	String	BSTR	8*	Due date in the format 'YYYYMMDD' or empty. Automatically calculated from OpDate and PaymentCondID if empty. Only applicable if AccountType equals atCustomer or atSupplier. Ignored for general account records.	'19981001' "	→ '19981001' → '19980923' (assigned from OpDate)
53	SettleNumber	String	BSTR	8*	Settlement number. You can settle amounts by creating different bookings with the same settlement number. The settlement number must be high enough for Wings in order not to collide with other, existing settlements. WAI will automatically add 50000000 to settlement numbers smaller than '50000000'.	" '123' '90000123'	→ automatically assigned → '50000123' → '90000123'
54	ReminderLevel	Longint	I4	-	Level number of the reminder already sent to a customer for this outstanding value. Only to be used with DocumentSort = dsOBCustomer. Default: 0; Maximum: 9	1	

	Property Name	Type	COM Type	Length	Description	Examples
Analytical Booking Detail (only used in case of manual analytical distribution or for purely analytical bookings)						
/WINGSACCOUNTING/SESSION/BOOKINGBATCH/BOOKING/DETAIL/ANALYTICALDETAIL						
OR						
/WINGSACCOUNTING/SESSION/BOOKINGBATCH/BOOKING/ANALYTICALDETAIL						
55	AnalDistribution	Longint	I4	-	Analytical distribution 1-5. Default is 1.	
56	AnalAccountID*	String	BSTR	8*	Analytical Account number . This must be an existing account. This is an obligatory property.	‘802113’
57	AnalAmountHC	Currency	CY	15,2	Analytical amount in the header currency. Only to be used if the base currency is not known. Not to be used for purely analytical bookings.	-112.10 (EUR)
58	AnalAmountBC	Currency	CY	15,2	Analytical amount in the base currency. If not filled, it will be calculated from the header currency.	-4522.00 (BEF) (-112.10 * 40.3399)
59	AnalQuantity	Real	R8	14,4	Quantity for analytical account .	5.00
60	AnalComment	String	BSTR	128	A comment for this booking line. This comment is visible on the printed journal. Only to be used for purely analytical bookings.	

	Property Name	Type	COM Type	Length	Description	Examples
61	AnalGenAccountID	String	BSTR	8*	General account number (default will be used if empty). Must be the same for all detail lines. Only to be used for purely analytical bookings.	
Intrastat Detail (optional) <i>/WINGSACCOUNTING/SESSION/BOOKINGBATCH/BOOKING/INTRASTATDETAIL</i>						
62	IntrFlowCode	Enum	I4	-	ifcArrival (1): Regime 19: Arrival ifcDispatch (2): Regime 29: Dispatch Defaults to Arrival or Dispatch according to the operation and document type.	1
63	IntrCountryID	String	BSTR	3*	Post office country code. Must be an EU country other than 'B'. Defaults to the country of the customer/supplier.	'F'
64	IntrCountryISO2	String	BSTR	2*	Two charactered ISO country code. The code must exist in the Wings country file.	'BE'
65	IntrRegion*	Longint	I4	-	Intrastat region.	3
66	IntrIncoterms(*)	String	BSTR	3*	Intrastat delivery code (Incoterms). Only mandatory for extended declarations.	'EXW'

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
67	IntrTransport(*)	Longint	I4	-	Intrastat transport code. Must be one of the following values (see intrastat manual): 1, 2, 3, 4, 5, 7, 8, 9 Only mandatory for extended declarations.	3
68	IntrTransaction*	Longint	I4	-	Intrastat transaction code. Must be one of the following values (see intrastat manual): 1, 2, 3, 4, 5, 7, 8, 9	1
69	IntrCN8Code*	String	BSTR	8*	Intrastat combined nomenclature code. Must be an existing intrastat code.	'84714190'
70	IntrWeight	Real	R8	13,2	Net mass in kg. Can be optional for certain intrastat codes.	216.00
71	IntrAdditional	Real	R8	13,2	Additional unit quantity (if necessary for intrastat code).	5
72	IntrAmountHC	Currency	CY	15,2	Statistical amount in the header currency. If not filled, IntrAmountBC must be used.	18540
73	IntrAmountBC	Currency	CY	15,2	Statistical amount in the base currency. If not filled, it will be calculated from the header currency.	459.59

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
General						
74	<i>ErrorCode</i>	Enum	I4	-	Error code. erNone (0): ... Is reset to erNone after every successful send.	erNone 2
75	<i>ErrorString</i>	String	BSTR	-	Error as descriptive text.	'Booking not in balance'
76	<i>ErrorProperty</i>	String	BSTR	-	Name of property that caused the error.	'LineAmountBC'

METHODS:

	<i>Method Name</i>	<i>Result Type</i>	<i>COM Type</i>	<i>Description</i>	<i>Special considerations</i>
1	SendHeader	Boolean	BOOL	<p>Sends the booking header to WAI. Must be the first method to call.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if the header has the correct format.</p> <p>False means an error condition. Check ErrorCode and ErrorString for a description of the error.</p>
2	SendAnalyticalDetail	Boolean	BOOL	<p>Sends an analytical distribution detail object to WAI. This call is only needed to manually distribute an amount onto several analytical accounts. This method is repeatedly called for each analytical booking line, before calling SendDetail. The analytical detail properties are reset.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if the analytical detail has the correct format.</p> <p>False means an error condition. Check ErrorCode and ErrorString for a description of the error.</p>

	<i>Method Name</i>	<i>Result Type</i>	<i>COM Type</i>	<i>Description</i>	<i>Special considerations</i>
3	SendDetail	Boolean	BOOL	<p>Sends the booking detail object to WAI. This method is repeatedly called for each booking line. The detail properties are reset.</p> <p>Not to be used for purely analytical bookings.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if the detail has the correct format.</p> <p>False means an error condition. Check ErrorCode and ErrorString for a description of the error.</p>
4	SendIntrastatDetail	Boolean	BOOL	<p>Sends the booking intrastat object to WAI. This method is repeatedly called for each intrastat line. The intrastat properties are reset. Can only be called for incoming and outgoing operations.</p> <p>Not to be used for purely analytical bookings.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if the detail has the correct format.</p> <p>False means an error condition. Check ErrorCode and ErrorString for a description of the error.</p>
5	AddAttachment Parameter: String	Boolean	BOOL	Adds a file as attachment to this booking. To be sent before SendBooking.	E.g. AddAttachment('C:\Docs\INV01254.PDF')

	<i>Method Name</i>	<i>Result Type</i>	<i>COM Type</i>	<i>Description</i>	<i>Special considerations</i>
6	SendBooking	Boolean	BOOL	<p>Sends the booking to WAI. This method is called after the last SendDetail call to test the complete booking. The header properties are reset.</p> <p>The bookings will only be registered in Wings after the SendBatch command.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if all booking checks are successful.</p> <p>False means an error condition. Check ErrorCode and ErrorString for a description of the error.</p>
7	SendBatch	Boolean	BOOL	<p>Registers the batch of bookings in Wings. This method should only fail due to locking conflicts (virtually all checks are already performed). If a problem occurs, SendBatch will return false and all actions performed will be rolled back. Unless you have called Clear, you can retry the SendBatch method in case of failure.</p> <p>In most cases, SendBatch can immediately follow SendBooking. Because of possible locking conflicts, it is advisable not to make the batch too big.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	<p>Returns true if all bookings were registered successfully.</p> <p>False means an error condition. Check ErrorCode and ErrorString for a description of the error.</p>
8	Clear	-	-	<p>Clears the batch and resets all properties. Also resets <i>ErrorCode</i> and <i>ErrorString</i>.</p>	<p>You need to call this method to restart sending bookings to WAI after an error.</p>

YJOURNALCLOSING

/WINGSACCOUNTING/SESSION/JOURNALCLOSING

PROPERTIES:

	Property Name	Type	COM Type	Length	Description	Examples
1	Connection	Object	OBJ	-	Connected YConnection object reference. You must set this before any other property.	
2	BookCode(*)	String	BSTR	4*	Code. WAI will convert to uppercase if necessary.	'V02'
3	BookType(*)	Enum	I4	-	btIncoming (1) btOutgoing (2) btFinancial (5)	btOutgoing 2
4	Period	String	BSTR	6	Period in the format 'YYYYMM'. The period must be in a valid range according to closed periods and open financial years.	'200906'
5	ErrorCode	Enum	I4	-	Error code. erNone (0): No error ... Is reset to erNone after every successful query.	erNotFound 1
6	ErrorString	String	BSTR	-	Error as descriptive text.	'Not found'
7	ErrorProperty	String	BSTR	-	Name of property that caused the error.	'BookCode'

METHODS:

	<i>Method Name</i>	<i>Parameters</i>	<i>Result Type</i>	<i>Description</i>	<i>Special considerations</i>
1	CloseJournal	-	Boolean	<p>Closes the journals indicated by either BookCode or BookType.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	False means an error condition. Check ErrorCode and ErrorString for a description of the error.

YINFO

PROPERTIES:

	Property Name	Type	COM Type	Length	Description	Examples
1	Connection	Object	OBJ	-	Connected YConnection object reference. You must set this before any other property.	
2	AccountType	Enum	I4	-	atGeneral (1): General account atCustomer (2): Customer account atSupplier (3): Supplier account	
3	AccountID*	String	BSTR	8*	General Account number or Customer or Supplier ID. This must be an existing account. This is an obligatory property.	<i>(general account)</i> '704000' '704000 ' <i>(customer/supplier)</i> '201' → '00000201'
Function return values						
4	ClosedPeriod	String	BSTR	6	Last closed period. Booking periods should be greater than this period.	'199901'
5	Amount	Currency	CY	15,2	Result amount in base currency. The base currency is taken from the financial year that includes ToPeriod (see below), or - if there is no ToPeriod - the current system date is used to determine the financial year.	

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
6	<i>CurrencyID</i>	String	BSTR	3	Currency of the amount. This is always the base currency for the financial year of the last period.	'EUR'
7	<i>Quantity</i>	Real	R8	14,4	Quantity if available. Only applicable to GetBalance().	5.00
8	<i>QuantUnit</i>	String	BSTR	3	Unit description of eventual quantity.	'kg'
9	<i>LastDate</i>	String	BSTR	8	Last settlement date. Only applicable to GetOutstanding().	'19990105'
Outstanding return values						
10	<i>SettleNumber</i>	String	BSTR	8	Settlement number of the outstanding record.	'70000051'
11	<i>BookCode</i>	String	BSTR	4	Book code of the original booking (invoice).	'V01'
12	<i>RunningNumber</i>	String	BSTR	6	Running number of the original booking (invoice).	'040000020'
13	<i>Period</i>	String	BSTR	6	Period of the original booking.	'200404'
14	<i>OpDate</i>	String	BSTR	8	Date of the original booking.	'20040401'
15	<i>DueDate</i>	String	BSTR	8	Due date of the outstanding record.	'20040501'
16	<i>DetAmountBC</i>	Currency	CY	15,2	Original amount in base currency.	2018.00
17	<i>DetAmountDC</i>	Currency	CY	15,2	Original amount in the original currency.	2018.00
18	<i>DetCurrencyID</i>	String	BSTR	3	Original currency.	'EUR'

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
19	<i>OutAmountBC</i>	Currency	CY	15,2	Amount still to be paid in the base currency.	2018.00
20	<i>OutAmountDC</i>	Currency	CY	15,2	Amount still to be paid in the original currency.	2018.00
21	<i>Comment</i>	String	BSTR	128	Comment of the original booking.	
22	<i>ReminderLevel</i>	Longint	I4	-	Reminder level (applicable only for customer records).	1
23	<i>ProForma</i>	Boolean	BOOL	-	True (-1) when outstanding record was booked via a pro forma customer/supplier. Otherwise False (0).	
General						
24	<i>ErrorCode</i>	Enum	I4	-	Error code. erNone (0): No error ... Is reset to erNone after every successful query.	erNone 2
25	<i>ErrorString</i>	String	BSTR	-	Error as descriptive text.	'Not Found'
26	<i>ErrorProperty</i>	String	BSTR	-	Name of property that caused the error.	'AccountID'

METHODS:

	Method Name	Parameters	Result Type	Description	Special considerations
1	GetBalance	FromPeriod: String ToPeriod: String	Boolean	<p>Returns the balance and eventual quantity for the designated account.</p> <p><i>FromPeriod</i> and <i>ToPeriod</i> are in the format 'YYYYMM' but can be left blank. If both are blank, the current balance is returned. If only <i>FromPeriod</i> is blank, the balance with transfer up to <i>ToPeriod</i> is returned.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	False means an error condition. Check <code>ErrorCode</code> and <code>ErrorString</code> for a description of the error.
2	GetOutstanding	SettleNumber: String	Boolean	<p>Returns the settled balance attached to <i>SettleNumber</i>.</p> <p>See <i>YBooking.SettleNumber</i> for more information.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	False means an error condition. Check <code>ErrorCode</code> and <code>ErrorString</code> for a description of the error.

	<i>Method Name</i>	<i>Parameters</i>	<i>Result Type</i>	<i>Description</i>	<i>Special considerations</i>
3	GetTurnover	FromPeriod: String ToPeriod: String	Boolean	<p>Returns the commercial turnover for the designated account.</p> <p><i>FromPeriod</i> and <i>ToPeriod</i> must be in the format 'YYYYMM'.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	False means an error condition. Check <i>ErrorCode</i> and <i>ErrorString</i> for a description of the error.
4	ChangeSettleNumber	SourceNumber: String DestNumber: String	Boolean	<p>Changes the settlement number from <i>SourceNumber</i> to <i>DestNumber</i>.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	False means an error condition. Check <i>ErrorCode</i> and <i>ErrorString</i> for a description of the error.
5	BlockOutstanding	SettleNumber: String	Boolean	<p>Blocks the payment of the invoice attached to <i>SettleNumber</i>.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	False means an error condition. Check <i>ErrorCode</i> and <i>ErrorString</i> for a description of the error.

	Method Name	Parameters	Result Type	Description	Special considerations
6	UnblockOutstanding	SettleNumber: String	Boolean	<p>Unblocks the payment of the invoice attached to <i>SettleNumber</i>.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	False means an error condition. Check ErrorCode and ErrorString for a description of the error.
7	FirstOutstanding	LastRecord: Boolean	Boolean	<p>Gets the first outstanding record for the designated account. If LastRecord contains True after calling this method, there are no more records. If there is no records for the designated account, all properties will contain empty values.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	False means an error condition. Check ErrorCode and ErrorString for a description of the error.
8	NextOutstanding	LastRecord: Boolean	Boolean	<p>Gets the next outstanding record for the designated account. If LastRecord contains True after calling this method, there are no more records.</p> <p>Result:</p> <p>True (-1): Success</p> <p>False (0): Error</p>	False means an error condition. Check ErrorCode and ErrorString for a description of the error.

YSQL

/WINGSACCOUNTING/SESSION/SQL

PROPERTIES:

	<i>Property Name</i>	<i>Type</i>	<i>COM Type</i>	<i>Length</i>	<i>Description</i>	<i>Examples</i>
1	Connection	Object	OBJ	-	Connected YConnection object reference. You must set this before any other property.	
2	SQLStatement*	String	BSTR	no practical limit	SQL SELECT statement to perform. This is an obligatory property.	select *from AdmRepr
3	ResultFile	String	BSTR	no practical limit	Filename of the result file (XML). If not used, Wings will give a temporary name and put the file in a temporary directory.	'C:\Temp\MyResult.XML'
4	<i>ErrorCode</i>	Enum	I4	-	Error code. erNone (0): No error ... Is reset to erNone after every successful query.	erInternalError 41
5	<i>ErrorString</i>	String	BSTR	-	Error as descriptive text.	'Internal error (see property)'
6	<i>ErrorProperty</i>	String	BSTR	-	Name of property that caused the error.	'EAdsDatabaseError...'

METHODS:

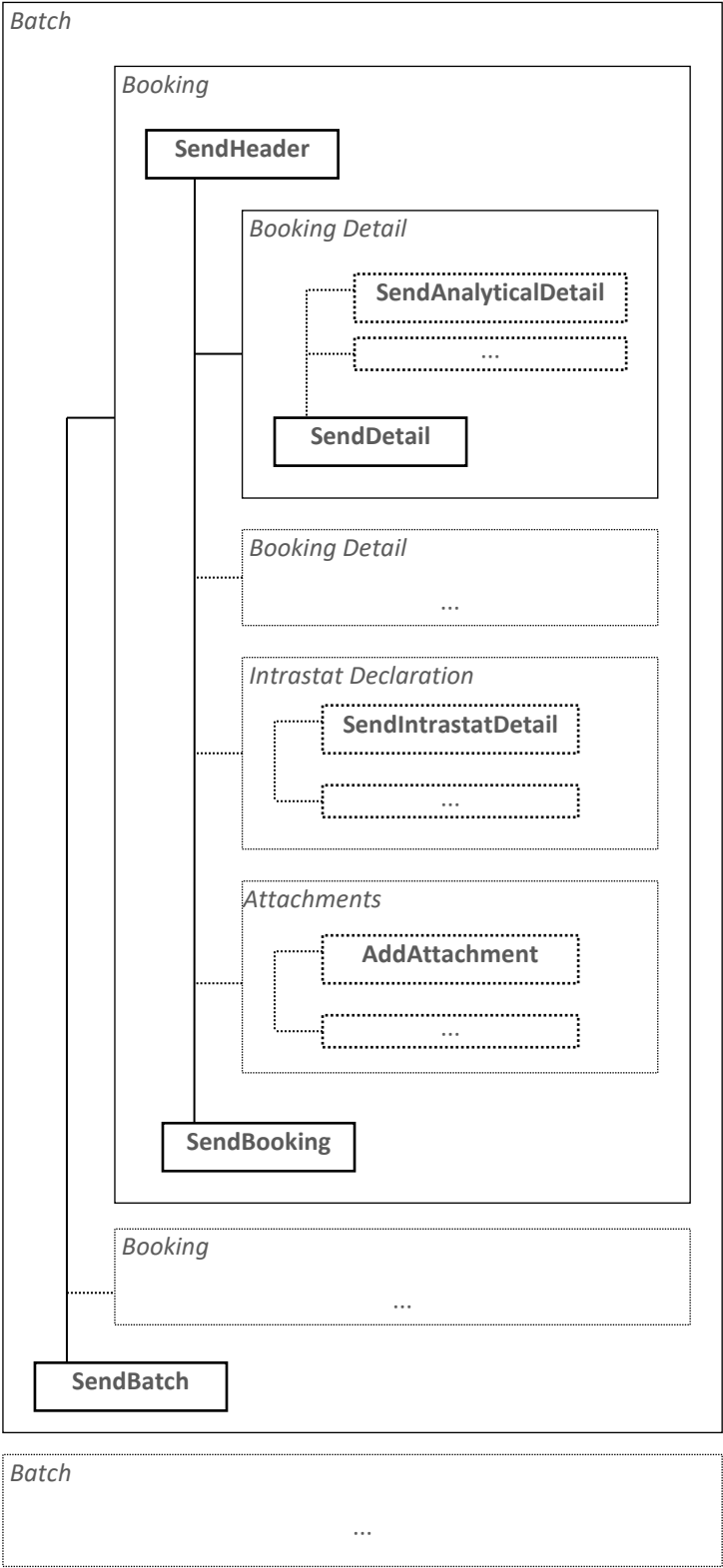
	<i>Method Name</i>	<i>Parameters</i>	<i>Result Type</i>	<i>Description</i>	<i>Special considerations</i>
1	ExecuteSQL	-	Boolean	Executes the query and puts the result in the XML file designated by the ResultFile property. Result: True (-1): Success False (0): Error	False means an error condition. Check ErrorCode and ErrorString for a description of the error.

ERRORCODE

<i>ErrorCode</i>	<i>Number</i>	<i>Description</i>	<i>ErrorProperty</i>	<i>Comment</i>
erNone	0	Successful completion	-	
erNotFound	1	Not found	normal property	Record not found in table
erNotConnected	2	Not connected	normal property	YConnection not connected
erInvalidValue	3	Invalid value	normal property	Property validation failed
erInvalidRange	4	Out of allowed range	normal property	Enum not within acceptable values, or Amount deviates too much of calculated counterpart
erInvalidStrLen	5	Incorrect string length	normal property	String too long (or too short)
erInvalidEmpty	6	May not be empty/zero	normal property	String or amount is required
erInvalidUsed	7	Must be empty/zero	normal property	Property may not be used in this situation
erInvalidSign	8	Invalid sign (+/-)	normal property	Amount/number may not be positive/negative
erInvalidDec	9	Invalid decimals	normal property	Amount has too many decimals for this currency
erInvalidFound	10	Already exists	normal property	Creation failed because record already exists
erInvalidBalance	11	Booking not balanced	normal property	Sum of LineAmountHC/DC was not 0 for booking or sum of AnalAmountDC/BC was not 0 for analytical distribution
erNotCalled	21	Method was not called	method name	Method calling order is wrong
erInvalidCalled	22	Method was already called	method name	Method calling order is wrong

<i>ErrorCode</i>	<i>Number</i>	<i>Description</i>	<i>ErrorProperty</i>	<i>Comment</i>
erInvalidVersion	23	Data need to be updated	DossierID	Data of this dossier is of an older version; open the dossier in Wings in order to update
erDossierLocked	31	Dossier is locked	normal property	A Wings user requires exclusive usage of dossier
erTableLocked	32	Table is locked	table name	Another user requires exclusive usage of table, or Exclusive usage of table cannot be granted
erRecordLocked	33	Record is locked	table name	Record already locked by another user
erInternalError	41	Internal error (see property)	actual error message	An exception occurred during processing

SCHEMATIC OVERVIEW FOR SENDING BOOKINGS



WAI.INI

You can put default values for properties in an INI file. This is very useful because it allows you to ignore values that you don't use in your program and still have control over which values are imported in Wings. Especially for crucial properties for Wings - like Customer.CentralizerID - that are rarely used in non-accounting programs, this is a handy feature. When the user wants to change the data in Wings, the default data can be adapted in the INI file, without the need to change any program.

LOCATION

Because the default values are dossier dependent, the INI file has to be put in the dossier's directory. E.g. *C:\Wings\Data\Main\WAI.INI* for single dossier systems. Alternatively, you can specify a different filename and/or location through the YConnection.IniFileName property.

STRUCTURE

Like every INI file, WAI.INI has a number of sections, each section containing keys and values. The section names are between square brackets and should contain the WAI object name. For the keys you should use the property names, immediately followed by an equal sign (=) and the value.

Following rules apply to the INI entries:

- do not use quotes for string values
- do not use additional spaces between the key, the equal sign and the value
- boolean values are considered False if zero, True if non-zero
- do not use thousand separators in numeric values
- for the YConnection object, you can only specify the RecordTimeOut and TableTimeOut properties

EXAMPLE

```
[YConnection]
RecordTimeOut=1500
TableTimeOut=1500

[YCustomer]
CentralizerID=400000
GenAccountID=701000
CurrencyID=EUR
GroupID=GEN
ShowDetail=1
PaymentCondID=F008
CreditLimit=10000
```