

SIF – Generic Web Service Layer

Table of Contents

1	About.....	5
1.1	Usage.....	5
1.2	Authentication and ADContextUser.....	5
1.3	Contact identifiers.....	6
1.4	Validation of personal id number and enterprise number.....	6
1.5	Code values.....	6
1.6	Common property tags.....	7
2	Contact service.....	8
2.1	SynchronizeContactPerson.....	8
2.1.1	Input.....	8
2.1.2	Output.....	9
2.2	SynchronizePrivatePerson.....	9
2.2.1	Input.....	10
2.2.2	Output.....	11
2.3	SynchronizeEnterprise.....	11
2.3.1	Input.....	13
2.3.2	Output.....	14
2.4	GetContactPersons.....	14
2.4.1	Input.....	14
2.4.2	Output.....	15
2.5	GetPrivatePersons.....	15
2.5.1	Input.....	15
2.5.2	Output.....	16
2.6	GetEnterprises.....	17
2.6.1	Input.....	17
2.6.2	Output.....	17
2.7	Sub Contracts.....	19
2.7.1	Data Contract: Address.....	19
2.7.2	Data Contract: ContactPersonBase.....	20
2.7.3	Data Contract: PrivatePersonBase.....	20

2.7.4	Data Contract: EnterpriseResult.....	20
3	User service.....	22
3.1	SynchronizeUser.....	22
3.1.1	Input.....	22
3.1.2	Output.....	23
3.2	GetUsers.....	23
3.2.1	Input.....	23
3.2.2	Output.....	23
3.3	Sub Contracts.....	24
3.3.1	Data Contract: UserBase.....	24
3.3.2	Data Contract: UserProfile.....	24
3.3.3	Data Contract: AccessGroupMembership.....	25
4	Case service.....	25
4.1	CreateCase.....	25
4.1.1	Input.....	25
4.1.2	Output.....	29
4.2	UpdateCase.....	30
4.2.1	Input.....	30
4.2.2	Output.....	33
4.3	GetCases.....	34
4.3.1	Input.....	34
4.3.2	Output.....	35
4.4	Sub Contracts.....	36
4.4.1	Data Contract: CaseContactParameter.....	36
4.4.2	Data Contract: ClassCodeParameter.....	36
4.4.3	Data Contract: CaseResult.....	37
4.4.4	Data Contract: AdditionalFieldParameter.....	38
5	Document service.....	39
5.1	CreateDocument.....	39
5.1.1	Input.....	39
5.1.2	Output.....	43
5.2	UpdateDocument.....	43
5.2.1	Input.....	43
5.2.2	Output.....	47
5.3	SignOffDocument.....	47

5.3.1	Input.....	47
5.3.2	Output.....	48
5.4	GetDocuments.....	49
5.4.1	Input.....	49
5.4.2	Output.....	50
5.5	Sub Contracts.....	50
5.5.1	Data Contract: DocumentContactParameter.....	50
5.5.2	Data Contract: UnregisteredContactParameter.....	51
5.5.3	Data Contract: ResponsibleEnterprise.....	52
5.5.4	Data Contract: ResponsiblePerson.....	52
5.5.5	Data Contract: DocumentResult.....	52
5.5.6	Data Contract: DocumentContactResult.....	55
5.5.7	Data contract: CreateFileFromTemplateParameter.....	56
5.5.8	Data Contract: File.....	56
6	File service.....	57
6.1	Upload.....	57
6.1.1	Input.....	57
6.1.2	Output.....	57
6.2	CheckOutAndGetFileForExternalControl.....	57
6.2.1	Input.....	58
6.2.2	Output.....	58
6.3	UploadAndCheckInFileFromExternalControl.....	58
6.3.1	Input.....	58
6.3.2	Output.....	59
7	Support methods.....	60
7.1	Ping.....	60
8	Dependencies to 360.....	60
8.1	PDF Converting.....	60
9	Web Service client responsibility.....	60
10	Common Code Table values.....	61
10.1	Case Status.....	61
10.2	Document Status.....	61
10.3	Document Category.....	62
10.4	Document Contact Role (Activity Contact role).....	63

1 About

This document describes the Web service layer of SI Integration Framework (SIF), which can be used by external systems to create, update and read data in 360.

Four types of methods are provided:

- Synchronize methods - Updates or creates depending on whether they exist or not.
- Create methods
- Update methods
- Get methods - Gets entities based on search criteria's.

1.1 Usage

There are multiple web service URL's. The exact URL depends on the server configuration, but in most cases they will be something like

[http://\[servername\]:8088/SI.WS.Core/SIF/ContactService.svc](http://[servername]:8088/SI.WS.Core/SIF/ContactService.svc)

.. or CaseService.svc, DocumentService.svc, UserService.svc, FileService.svc

WSDL's for these services are provided along with this document. WSDL specifies the functionality of a web service, and can be used to build the client code.

The default service endpoints use Soap 1.1, conform to WS-I Basic Profile 1.2, and use Basic authentication. Each service (apart from FileService) also has alternative endpoints with different configurations. These are accessed by adding a postfix to the URL.

Endpoint	Soap	Authentication	Other
.svc	1.1	Basic	
.svc/Basic	1.1	Basic	
.svc/Ntlm	1.1	Ntlm	
.svc/BasicMtom	1.1	Basic	Mtom
.svc/NtlmMtom	1.1	Ntlm	Mtom

Other authentication methods and Soap 1.2 can be configured if required.

1.2 Authentication and ADContextUser

The ADContextUser property is supported by all the web service methods, and contains the login name of a valid 360 user. If provided, the operation will execute as (impersonate) this user. If not provided, the operation will execute as the user that has authenticated against the web service.

Only some users are allowed to impersonate other users. For instance, if the client authenticates against the service with the user "domain\administrator", and specifies "domain\user1" as ADContextUser, "domain\administrator" must have impersonation rights in 360. This is specified in the configuration file of the web service.

Typical scenario 1: The client of the web service is Application X. A service user is created for this application, "domain\appx_svc", which is configured with impersonation rights. The client

authenticates with “domain\appx_svc”, but provides the login of the end user, “domain\user1”, as ADContextUser. The operation is then executed as user1. For instance, if a case is created, user1 will be logged as the user who created it.

Typical scenario 2: Same as 1, except no ADContextUser is provided. In this case, the operation will be executed as appx_svc. If a contact is created, appx_svc will be logged as the user who created it.

It is recommended that a dedicated service user is created either for all clients to use in common (“domain\360sif_svc”), or for each of them to use separately (“appx_svc”, “appy_svc”, ..). If ADContextUser is provided, it’s enough with one common service user. If ADContextUser is not provided, it is recommended to have a service user for each client, so that the 360 log shows where the data came from. For instance, if both Application X and Application Y create contacts in 360, it is useful to know which of them created a specific contact.

1.3 Contact identifiers

There are three types of registered contacts in 360: Enterprises (/companies/departments), private persons and contact persons. A contact person is associated with an enterprise, a private person is not. A user always has a contact person, which contains the contact metadata of that user.

In the SIF web service, contacts are identified by their PersonalIdNumber (“fødselsnummer”) if they’re private persons, EnterpriseNumber (“organisasjonsnummer”) if they’re enterprises, or ExternalId if they’re contact persons. For instance, if you’re synchronizing a private person with PersonalIdNumber 1234567890, the service will look for a contact with that id, and update it if it exists.

When multiple external systems work with 360 contacts, or when there are existing contacts that should be referred to / updated by the web service, care must be taken to prevent duplicates and other problems. This should be analyzed before the system is put into production.

1.4 Validation of personal id number and enterprise number

When synchronizing private persons and enterprises, the service can optionally validate that the values comply with Norwegian fødselsnummer and organisasjonsnummer formats (11 and 9 digits, and satisfying a modulus 11 validation).

This functionality is toggled by the “SI.Data.WS.ValidateContactId” appSetting in the SIF web.config file. By default, it is set to false.

1.5 Code values

Some methods accept input in the form of code table values. For instance, when creating a case, you can set its Status property to a value from the Case status code table, such as “B”.

All properties that accept code values can alternatively accept the recno of the code, on the format “recno:123”. It is less risky to specify the code by recno, since it’s possible (but rare) for code values to change.

Read more about this in the [Common Code Table: values chapter](#).

1.6 Common property tags

(*)

This tag behind a property name means that the property is mandatory.

(\)

This tag behind property name means that at least one of properties with this symbol needs to be supplied.

2 Contact service

Typical URL: [http://\[servername\]:8088/SI.WS.Core/SIF/ContactService.svc](http://[servername]:8088/SI.WS.Core/SIF/ContactService.svc)

2.1 SynchronizeContactPerson

Creates or updates a contact person contact in 360 based on ExternalId. If the supplied ExternalId does not exist in 360 the contact is created. If the supplied ExternalId exists in 360 the contact is updated to mirror the supplied information.

In the scenario where there exists several contacts with the same ExternalId the oldest (lowest recno) will be updated, while an error message is logged to TraceView and the Windows Event Viewer. It is not possible that this web service method itself creates a duplicate of a contact, given that the ExternalId is correct.

OBS: If a property is not set, meaning it is blank or null; the corresponding value in 360 will be set to an empty value. This behavior is unique to the contact related synchronization methods.

2.1.1 Input

Data Contract: SynchronizeContactPersonParameter

Input Property	360° field	Comment
ADContextUser	-	Read the general information section.
FirstName(*)	Contact – Name2	Mandatory when creating the contact
MiddleName	Contact – Name2	Middle name of the contact, concatenated to FirstName.
LastName(*)	Contact – Name3	Mandatory when creating the contact
ExternalId(*)	Contact – ExternalID	The external id number of the contact used as a unique identifier for the contact in 360. If unique, a new contact is created, otherwise the contact is updated. If more than one external system use the web service, external id's must be unique also between external systems.
Enterprise(*)	Contact – ToEmployer	The OrgNo of the enterprise the contact is connected to, most likely as an employee.
Categories	Contact – Categories	List of categories, identified by their code name in the configured language, or by recno on the format "recno:123". If this property is not specified (null), no changes will be made to the user's existing categories.
Email	Contact – TeleObject 6	

PhoneNumber	Contact – TeleObject 1	
MobilePhone	Contact – TeleObject 27	
PrivateAddress	Contact - Addresses	PrivateAddress is an address object, see fields needed for input under “ Data Contract: Address ”.
PostAddress	Contact - Addresses	PostAddress is an address object, see fields needed for input under “ Data Contract: Address ”.
DataSource		DataSource is presented as a log entry under the Log-tab on the contact. The point is to track which systems have made updates to the contact. No logging is done if this property is empty or not set.
Gender	Contact - Gender	Enum Gender.Female or Gender.Male
Initials	Contact - Initials	<p>After DLL version 1.0.0.58408: Added. Null values will clear stored data.</p> <p>Before DLL version 1.0.0.58408: Not available and would not clear stored data.</p>
CustomNo1 CustomNo2 CustomNo3	Contact – No1 Contact – No2 Contact – No3	<p>Standard fields that can be used to store custom data in. Usage vary from customer to customer.</p> <p>Contrary to other properties on this contract, null values will NOT clear stored data. To clear stored data you have to input an empty “ ” string. Because of backwards compatibility considerations.</p>

2.1.2 Output

Data Contract: SynchronizeContactPersonResult

Output Property	360° field	Comment
Recno	Contact – Recno	Unique ID in 360 for the contact
Successful	-	If true, everything is ok. If false, the operation failed.
ErrorMessage	-	If unsuccessful, this contains a short error message
ErrorDetails	-	If unsuccessful, this contains a full stack trace

2.2 SynchronizePrivatePerson

This creates or updates a private person contact in 360 based on PersonalIdNumber. If the supplied PersonalIdNumber does not exist in 360 the contact is created. If the supplied PersonalIdNumber exists in 360 the contact is updated to mirror the supplied information.

In the scenario where there exists several contacts with the same PersonalIdNumber the oldest (lowest recno) will be updated, while an error message is logged to TraceView and the Windows Event Viewer. It is not possible that this web service method itself creates a duplicate of a contact, given that the PersonalIdNumber is correct.

OBS: If a property is not set, meaning it is blank or null; the corresponding value in 360 will be set to an empty value. This behavior is unique to the contact related synchronization methods.

2.2.1 Input

Data Contract: SynchronizePrivatePersonParameter

Input Property	360° field	Comment
ADContextUser	-	Read the general information section.
FirstName(*)	Contact – Name2	Mandatory when creating the contact
MiddleName	Contact – Name2	Middle name of the contact, concatenated to FirstName.
LastName(*)	Contact – Name3	Mandatory when creating the contact
PersonalIdNumber(*)	Contact – Referencenumber	The personal identity number of the contact used as a unique identifier for the contact in 360. If unique, a new contact is created, otherwise the contact is updated. Can optionally validate that the values comply with Norwegian fødselsnummer and organisasjonsnummer formats (11 and 9 digits, and satisfying a modulus 11 validation).
Email	Contact – TeleObject 6	
PhoneNumber	Contact – TeleObject 1	
MobilePhone	Contact – TeleObject 27	
PrivateAddress	Contact - Addresses	PrivateAddress is an address object, see fields needed for input under " Data Contract: Address ".
PostAddress	Contact - Addresses	PostAddress is an address object, see fields needed for input under " Data Contract: Address ".

DataSource		DataSource is presented as a log entry under the Log-tab on the contact. The point is to track which systems have made updates to the contact. No logging is done if this property is empty or not set.
Categories	Contact – Categories	List of categories, identified by their code name in the configured language, or by recno on the format “recno:123”. If this property is not specified (null), no changes will be made to the contact’s existing categories.
Gender	Contact - Gender	Enum Gender.Female or Gender.Male
Initials	Contact - Initials	After DLL version 1.0.0.58408 : Added. Null values will clear this field. Before DLL version 1.0.0.58408 : Not available and was never cleared.
CustomNo1 CustomNo2 CustomNo3	Contact – No1 Contact – No2 Contact – No3	Standard fields that can be used to store custom data in. Usage vary from customer to customer. Contrary to other properties on this contract, null values will NOT clear stored data. To clear stored data you have to input an empty “ ” string. Because of backwards compatibility considerations.

2.2.2 Output

Data Contract: SynchronizePrivatePersonResult

Output Property	360° field	Comment
Recno	Contact – Recno	Unique ID in 360 for the contact
Successful	-	If true, everything is ok. If false, the operation failed.
ErrorMessage	-	If unsuccessful, this contains a short error message
ErrorDetails	-	If unsuccessful, this contains a full stack trace

2.3 SynchronizeEnterprise

This creates or updates an enterprise contact in 360 based on PersonalIdNumber. If the supplied PersonalIdNumber does not exist in 360 the contact is created. If the supplied PersonalIdNumber exists in 360 the contact is updated to mirror the supplied data.

In the scenario where there exists several contacts with the same PersonalIdNumber the oldest (lowest recno) will be updated, while an error message is logged to TraceView and the Windows Event Viewer. It is not possible that this web service method itself creates a duplicate of a contact, given that the PersonalIdNumber is correct.

OBS: If a property is not set, meaning it is blank or null; the corresponding value in 360 will be set to an empty value. This behavior is unique to the contact related synchronization methods.

2.3.1 Input

Data Contract: SynchronizeEnterpriseParameter

Input Property	360° field	Comment
ADContextUser	-	Read the general information section.
Name(*)	Contact – Name	Mandatory when creating the contact
EnterpriseNumber(*)	Contact – Referencenumber	Enterprise number of the company. Represents a unique identifier for the enterprise in 360. Can optionally validate that the values comply with Norwegian fødselsnummer and organisasjonsnummer formats (11 and 9 digits, and satisfying a modulus 11 validation).
ParentEnterpriseNumber	Contact – ToEmployer	Enterprise number of the parent company.
PhoneNumber	Contact – Teleobject 1	
Email	Contact – Teleobject 6	
PostAddress	Contact – Addresses	Postal address of the company. PostAddress is an address object, see fields needed for input under chapter: “Data Contract: Address” .
OfficeAddress	Contact – Addresses	Office address of the company. OfficeAddress is an address object, see fields needed for input under chapter: “Data Contract: Address” .
DataSource		DataSource is presented as a log entry under the Log-tab on the contact. The point is to track which systems have made updates to the contact. No logging is done if this property is empty or not set.
Categories	Contact – Categories	List of categories, identified by their code name in the configured language, or by recno on the format “recno:123”. If this property is not specified (null), no changes will be made to the contact’s existing categories.
ContactRelations	Contact - FromMeToOthers	List of contact – contact relations. All relations are hardcoded to relation role “Cooperate with”. Note: Only a private persons or an enterprise may be added as a relation to a Contact.
ReferenceNumber	Contact – FromMeToOthers - Referencenumber	EnterpriseNumber or PersonalIdNumber of the connected contact.

Initials	Contact - Initials	After DLL version 1.0.0.58408 : Added. Null values will clear this field. Before DLL version 1.0.0.58408 : Not available and was never cleared.
CustomNo1 CustomNo2 CustomNo3	Contact – No1 Contact – No2 Contact – No3	Standard fields that can be used to store custom data in. Usage vary from customer to customer. Contrary to other properties on this contract, null values will NOT clear stored data. To clear stored data you have to input an empty “ ” string. Because of backwards compatibility considerations.

2.3.2 Output

Data Contract: SynchronizeEnterpriseResult

Output Property	360° field	Comment
Recno	Contact - Recno	Unique ID in 360 for the contact
Successful	-	If true, everything is ok. If false, the operation failed.
ErrorMessage	-	If unsuccessful, this contains a short error message
ErrorDetails	-	If unsuccessful, this contains a full stack trace

2.4 GetContactPersons

Retrieves contact persons from 360 based on criteria defined in the input parameter. At least one criteria (Name, Recno, Categories or ExternalId) must be given. Returns a list of matching contact persons.

2.4.1 Input

Data Contract: GetContactPersonParameter

Input Property	360° field	Comment
ADContextUser		Read the general information section.
Name(\)	Contact – SearchName	Name of the contact. Returns all contacts with a name that contains the search name. Example: “Inge” will return “Knut Inge N.” and “Inge Larsen”
ExternalId(\)	Contact – ExternalID	External identity number (unique

		identifier) of the contact. Must be an exact match to return a contact.
Recno(\)	Contact – Recno	Recno (unique identifier in 360) of the contact. Must be an exact match to return a contact.
Categories(\)	Contact – Categories	List of categories, identified by their code name in the configured language, or by recno on the format “recno:123”.

2.4.2 Output

Data Contract: GetContactPersonResult

Output Property	360° field	Comment
ContactPersons	Contact	List of ContactPersonBase objects. See chapter “ Data Contract: ContactPersonBase ” for details.
Successful	-	Will be true as long as no exception occurred during the execution. This means that successful will be true even if no contact persons matching the criteria was found.
ErrorMessage	-	If successful is false, this contains a short error message
ErrorDetails	-	If successful is false, this contains a full stack trace

2.5 GetPrivatePersons

Retrieves private persons from 360 based on criteria defined in the input parameter. At least one criteria (Name, Recno or PersonalIdNumber) must be given. Returns a list of matching private persons.

2.5.1 Input

Data Contract: GetPrivatePersonParameter

Input Property	360° field	Comment
ADContextUser		Read the general information section.
Name(\)	Contact – SearchName	Name of the contact. Returns all contacts with a name that contains the search name. Example: “Inge” will return “Knut Inge N.” and “Inge Larsen”
PersonalIdNumber(\)	Contact –	Personal identity number (unique

	ReferenceNumber	identifier) of the contact. Must be an exact match to return a contact.
Recno()	Contact – Recno	Recno (unique identifier in 360) of the contact. Must be an exact match to return a contact.

2.5.2 Output

Data Contract: GetPrivatePersonResult

Output Property	360° field	Comment
PrivatePersons	Contact	List of PrivatePersonBase objects. See chapter “ Data Contract: PrivatePersonBase ” for details.
Successful	-	Will be true as long as no exception occurred during the execution. This means that successful will be true even if no private persons matching the criteria was found.
ErrorMessage	-	If successful is false, this contains a short error message
ErrorDetails	-	If successful is false, this contains a full stack trace

2.6 GetEnterprises

Retrieves enterprises from 360 based on criteria defined in the input parameter. At least one criteria (Name, Recno, Categories or EnterpriseNumber) must be given. Returns a list of matching enterprises.

2.6.1 Input

Data Contract: GetEnterpriseParameter

Input Property	360° field	Comment
ADContextUser	-	Read the general information section.
Name(\)	Contact – SearchName	Name of the company. Returns all contacts with a name that contains the search name. Example: "Innovation" will give a hit on "Software Innovation".
EnterpriseNumber(\)	Contact – ReferenceNumber	Enterprise identity number (unique identifier) of the company. Must be an exact match to return a contact
Recno(\)	Contact – Recno	Recno (unique identifier in 360) of the company. Must be an exact match to return an enterprise
Categories(\)	Contact – Categories	List of categories, identified by their code name in the configured language, or by recno on the format "recno:123".

2.6.2 Output

Data Contract: GetEnterpriseResult

Output Property	360° field	Comment
Enterprises	Contact	List of PrivatePersonBase objects. See the chapter " Data Contract: EnterpriseResult " for details.
Successful	-	Will be true as long as no exception occurred during the execution. This means that successful will be true even if no enterprises matching the criteria was found.
ErrorMessage	-	If successful is false, this contains a short error message
ErrorDetails	-	If successful is false, this contains a full stack trace

2.7 Sub Contracts

This chapter defines data contracts in use by the main data contracts of the Contact methods.

2.7.1 Data Contract: Address

Property	360° field	Comment
StreetAddress	Address - Address	Example: "Drammensveien 1"
ZipCode	Address - ZipCode	Zip code of address.
ZipPlace	Address - ZipPlace	Zip place (City) of address.
Country(*)	Address - Country	Country of address. To create an address the user has to include "Country" together with another field. E.g. just sending in "Country" will not create an address, but sending in Country and StreetAddress will create an address. Example of input formats: "Norway" or "NOR".
County	Address - County	Municipality of address. Norwegian: "Kommune". Examples of input formats: "BODØ" or "1804", it is recommended that the number format is used, i.e. "1804".
Area	Address - Area	County of address. Norwegian: "Fylke". Examples of input formats: "NORDLAND" or "18", it is recommended that the number format is used, i.e. "18".
State	Address - State	Very optional, only used for some foreign countries.

2.7.2 Data Contract: ContactPersonBase

Property	360° field	Comment
FirstName	Contact – Name2	
LastName	Contact – Name3	Mandatory when creating the contact
ExternalId	Contact – ExternalId	The external identity number of the contact, used as the unique identifier for the contact in 360.
Enterprise	Contact – ToEmployer	The OrgNo of the Enterprise the contact is connected to.
Email	Contact – TeleObject 6	
PhoneNumber	Contact – TeleObject 1	
MobilePhone	Contact – TeleObject 27	
PrivateAddress	Contact - Addresses	PrivateAddress is an address object, see details under the chapter: “Data Contract: Address” .
PostAddress	Contact - Addresses	PostAddress is an address object, see details under the chapter: “Data Contract: Address” .

2.7.3 Data Contract: PrivatePersonBase

Property	360° field	Comment
FirstName	Contact – Name2	
LastName	Contact – Name3	Mandatory when creating the contact
PersonalIdNumber	Contact – Referencenumber	The personal identity number of the contact, used as the unique identifier for the contact in 360.
Email	Contact – TeleObject 6	
PhoneNumber	Contact – TeleObject 1	
MobilePhone	Contact – TeleObject 27	
PrivateAddress	Contact - Addresses	PrivateAddress is an address object, see details under the chapter: “Data Contract: Address” .
PostAddress	Contact - Addresses	PostAddress is an address object, see details under the chapter: “Data Contract: Address” .

2.7.4 Data Contract: EnterpriseResult

Output Property	360° field	Comment
-----------------	------------	---------

EnterpriseNumber	Contact – Referencenumber	Unique ID in 360 for the contact, also the contact's personal ID number
Name	Contact – Name	
PhoneNumber	Contact – Teleobject 1	
Email	Contact – Teleobject 6	
OfficeAddress	Contact – Addresses	OfficeAddress is an address object, see details under the chapter: “Data Contract: Address”
PostAddress	Contact - Addresses	PostAddress is an address object, see details under the chapter: “Data Contract: Address”
ContactRelations	Contact – ToContacts	A list of contacts associated with the enterprise. The list contains EnterpriseContactResult objects.
Name	Contact – ToContacts - Name	Name of a contact person

3 User service

Typical URL: [http://\[servername\]:8088/SI.WS.Core/SIF/UserService.svc](http://[servername]:8088/SI.WS.Core/SIF/UserService.svc)

3.1 SynchronizeUser

Creates or updates a user in 360, identified by its login name. If the user does not exist, it is created. If it exists, it is updated to match the input.

A user must be connected to an existing 360 contact person, so always call SynchronizeContactPerson in the contact service first.

3.1.1 Input

Data Contract: SynchronizeUserParameter

Input Property	360° field	Comment
ADContextUser	-	Read the general information section.
Login (*)	User – UserID	Login name
ContactExternalId (*)	User – ToContact	Reference to the user's contact person
IsActive	User – Active	True/false. Default: True
Profiles	User profile	If the parameter is not specified (null), profiles are not synchronized, and no existing profiles will be overwritten. If it is specified, even with an empty list, existing profiles will be overwritten.
Role (*)	User profile – ToSysrole	Code from the Sysroles code table, (using the configured language), or recno on the format "recno:123"
Enterpriseld (*)	User profile – ToOrgUnit	The role's organization unit. Set to either it's external id or enterprise number, or recno on the format "recno:123"
FromDate	User profile – FromDate	A date the profile is valid from
ToDate	User profile – ToDate	A date the profile is valid to
AccessGroups	Access Group Members	If the parameter is not specified (null), access groups are not synchronized, and no existing group memberships will be overwritten. If it is specified, even with an empty list, existing group memberships will be overwritten.
AccessGroup (*)	AGM – ToAccessGroup	Name of the access group, or recno on the format "recno:123"
FromDate	AGM – FromDate	A date the membership is valid from

ToDate	AGM – ToDate	A date the membership is valid to
---------------	--------------	-----------------------------------

The Profiles and AccessGroups properties are optional. If not specified at all (null), no changes will be made to existing profiles and group memberships, allowing these to be maintained manually.

3.1.2 Output

Data Contract: SynchronizeUserResult

Output Property	360° field	Comment
Recno	User – Recno	Internal 360 id for the user
Successful		If true, everything is ok. If false, the operation failed.
ErrorMessage		If unsuccessful, this contains a short error message
ErrorDetails		If unsuccessful, this contains a full stack trace

3.2 GetUsers

Retrieves users from 360 based on criteria defined in the input parameter, input parameter can be empty. Possible criterias (UserId,ContactExternalId or none). Returns a list of matching users.

3.2.1 Input

Data Contract: GetUsersParameters

Input Property	360° field	Comment
ADContextUser		Read the general information section.
UserId	User - UserId	The ID that identifies the User (typically "domain\user").
ContactExternalId		Search users associated with the given contact external id

3.2.2 Output

Data Contract: GetUsersResult

Output Property	360° field	Comment
Users	User	List of User Base objects. See chapter "Data Contract - UserBase" for details.

3.3 Sub Contracts

This chapter defines data contracts in use by the main data contracts of the User methods.

3.3.1 Data Contract: UserBase

Property	360° field	Comment
Login		
ContactExternalId	Contact - ExternalID	Contact ID in the external system. If this user don't have an external id, the format will be "recno:xxxx"
IsActive		Boolean : Indicates if the user is active or not
Profiles	UserProfile	Array of connected user profiles. See chapter "Data Contract - UserProfile" for details.
AccessGroups	AccessGroup	Array of connected access groups. See chapter "Data Contract – AccessGroupMembership" for details.

3.3.2 Data Contract: UserProfile

Property	360° field	Comment
Role	UserProfile - ToSysrole	A relation to the System Role this profile is using.
Enterpriseld		External ID for user profile's organization unit.
FromDate	UserProfile - FromDate	The Start date that this profile is valid from.
Todate	UserProfile - ToDate	The profile is valid to this date.

3.3.3 Data Contract: AccessGroupMembership

Property	360° field	Comment
AccessGroup	AccessGroup - Code	Ident of this AccessGroup
FromDate	AccessGroup - FromDate	The Start date this accessgroup is valid from.
Todate	AccessGroup - ToDate	The accessgroup is valid to this date.

4 Case service

Typical URL: [http://\[servername\]:8088/SI.WS.Core/SIF/CaseService.svc](http://[servername]:8088/SI.WS.Core/SIF/CaseService.svc)

4.1 CreateCase

Creates a case in 360 with the case type Case ("Sak/Noark Sak" in Norwegian). The parameters in the data contracts accept either code value of the respective code or recno if nothing else is specified.

4.1.1 Input

Data Contract: CreateCaseParameter

Input Property	360° field	Comment
ADContextUser		Read the general information section.
Title(*)	Case – Description	
UnofficialTitle	Case – UnofficialTitle	If not set, Title will be used
ExternalId		External ID for the case.
Id	ExternalObject – Id	External id. String value.
Type	ExternalObject - ToExternalType	<p>External id type. Not mandatory. If not provided, default type "SI WS Case" will be used.</p> <p>External id is unique within each type, so type can be used if cases are importet from more than one external system which use ExternalId.</p> <p>Should have a name to reflect system and entity, e.g. "Visma Case".</p> <p>Types will be added to the code table <i>External type</i> when a new</p>

		type is provided.
StartDate	Case – StartDate	Start date of the case progress. Defaults to the creation date of the case.
Status	Case – ToCaseStatus	<p>Code value or recno. View common values.</p> <p>Example of use: Status="B"; sets the status to "Under handling"</p> <p>ArchiveCode and ArchiveType needs to be supplied to set the status that closes the case. E.g. "A" – "Avsluttet"</p> <p>Defaults according to the ADUserContext user's role:</p> <ul style="list-style-type: none"> - Archivist = Closed - Case Manager = Reserved
ClosedDate	Case – ClosedDate	<p>The date the case is closed. Should be used together with a closing status and is ignored otherwise.</p> <p>The following happens if the Status property is set to a closing status:</p> <ul style="list-style-type: none"> - If this property is set, the case will gain this date as ClosedDate. - If this property is not set 360 automatically sets the ClosedDate to the current date.
JournalUnit	Case – ToJournalUnit	Can be code value or recno. E.g. JournalUnit="Sentralarkiv"
SubArchive	Case – ToSubArchive	Can be code value or recno. E.g. SubArchive="Sakarkiv", "60001" or "recno:60001"
ArchiveCodes	Case – ToClassCodes	At least one ArchiveCode needs to be set to close the case. This is a list of ClassCodeParameter objects. See the chapter " Data Contract ClassCodeParameter " for details.
FiledOnPaper	Case – Paper	Indicating (true/false) if the case is stored on paper.

Keywords	Case – Keywords	List of keywords connected to the case. If a keyword that does not exist in 360 is used, this is added to the keywords code table (the user needs Admin rights to create keywords).
AccessCode	Case – ToAccessCode	Code value or recno. E.g. AccessCode = “U” or “recno:0” Some access codes require Paragraph to be specified as well.
Paragraph	Case - Paragraph	Can be code value or recno. E.g. Paragraph = “OFL § 11” Mandatory for some access codes.
AccessGroup	Case – ToAccessGroup	Can be code value or recno. E.g. AccessGroup = “Public”
Notes	Case – Notes	Notes on the case
CaseType	Case - CaseType	Code or recno value, for instance “Sak” or “recno:2”. Default: 2 (Noark case) Refers to the <i>Case type</i> code table.
ResponsiblePersonIdNumber	Case – OurRef	The external ID of the responsible person. Looks for a contact with ExternalId equal to the property input. It is only possible to set a contactperson in the same enterprise hierarchy as the ADUserContext user as responsible. Provide only ONE of the four possible fields to ID responsible person.
ResponsiblePersonRecno	Case – OurRef	Looks for a contact with Recno equal to the property input.
ResponsiblePersonEmail	Case – OurRef	Looks for a contact with Email equal to the property input.
ResponsiblePersonUserId	Case – OurRef	Looks for a contact with UserId (username, E.g. “no\name.name”)

ResponsibleEnterpriseNumber	Case – ToOrgUnit	<p>The enterprise number of the responsible enterprise.</p> <p>Looks for a contact with ExternalID first, then for one with reference number equal to the property input.</p> <p>If neither this nor the ResponsibleEnterpriseRecno property is set, the responsible enterprise will be set to the responsible person's enterprise. If the responsible person is not set, the responsible enterprise will be set to the main enterprise of all the users.</p> <p>It is only possible to set the responsible enterprise to an enterprise in the same enterprise hierarchy as the ADUserContext user.</p> <p>Provide only ONE of the two possible fields to ID responsible enterprise.</p>
ResponsibleEnterpriseRecno	Case – ToOrgUnit	Looks up enterprise based on recno. Must be recno of a valid enterprise. See also ResponsibleEnterpriseNumber.
Contacts	Case - ToCaseContact	List of contacts with associated roles. See sub contract CaseContactParameter
ReferringCases	Case – ReferringCases	List of case numbers of referring cases. E.g. ("14/0001", "14/0002") Ignored if empty
Project	Case – Project	Project number of the project this case should be connected to.
Remarks	Case – ToNotes	List of remarks to be added to case
Title (*)	Note – Title	Remark title Mandatory
Content	Note - Notes	Remark content
RemarkType (*)	Note - ToNoteType	Uses code table: Note type Can be code value or recno. Mandatory

ScrapCode	Case – ToScrapCode	Can be code value or recno. E.g. ScrapCode = "B"
Category	Case – ToCaseCategory	Can be code value or recno. E.g. Category = "Leveranse" or "recno:1"
AdditionalFields	-	Additional fields – see chapter for AdditionalFieldParameter for details, what kind of fields and how to use this parameter.

4.1.2 Output

Data Contract: CaseOperationResult

Output Property	360° field	Comment
Recno	Case - Recno	Unique ID in 360 for the case.
CaseNumber	Case - Name	The case number given by 360.
Successful	-	If true, everything is ok. If false, the operation failed.
ErrorMessage	-	If unsuccessful, this contains a short error message
ErrorDetails	-	If unsuccessful, this contains a full stack trace

4.2 UpdateCase

Updates a case in 360. If the case has a closed status this operation can only be done by an archivist.

If a property is set to null, meaning not provided in the parameter passed to the method, the property is not updated. To clear the field in 360 the property has to be explicitly set to "". Only works for string properties. Archive codes supplied to this method is ignored and no changes to an archive code is possible once the case is created.

4.2.1 Input

Data Contract: UpdateCaseParameter

Input Property	360° field	Comment
ADContextUser	-	Read the general information section.
CaseNumber(\)	Case – Name	The case number. This property will be used to determine which case will be updated. A case cannot have its case number updated.
ExternalId(\)		<p>External id. This property will be used to determine which case will be updated. A case cannot have its external ID updated.</p> <p>Lookup is based on the combination of Id and Type.</p> <p>Provide only ONE of the two possible fields to ID the case.</p>
Id	ExternalObject – Id	External id.
Type	ExternalObject - ToExternalType	If not provided, default value "SIF WS Case" will be used. This is the type set via CreateCase if type was not provided when importing a case.
Title	Case – Description	
UnofficialTitle	Case – UnofficialTitle	
StartDate	Case – StartDate	The start date of the case progress.
Status	Case – ToCaseStatus	<p>May only be set to a closing status if ArchiveCode and ArchiveType are also set or have been set.</p> <p>If the status is set to a closing status (e.g. "Closed") the closed date of the case is set to the current date.</p>

JournalUnit	Case – ToJournalUnit	Can be code value or recno. E.g. JournalUnit="Sentralarkiv"
SubArchive	Case – ToSubArchive	Can be code value or recno. E.g. SubArchive="Sakarkiv"
FiledOnPaper	Case – Paper	Indicating (true/false) if the case is stored on paper.
ArchiveCodes	Case – ToClassCodes	At least one ArchiveCode needs to be set to close the case. This is a list of ClassCodeParameter objects. Codes are appended. Existing codes can not be modified. See the chapter " Data Contract ClassCodeParameter " for details.
Keywords	Case – Keywords	When updating keywords this is the update rules: 1) New keywords are added 2) Existing keywords are ignored 3) Missing keywords are ignored
AccessCode	Case – ToAccessCode	Can be code value or recno. E.g. AccessCode = "U" If the access code requires a paragraph the field paragraph has to offer a valid paragraph. N.B: If a recno value is used, this recno value <u>can not</u> also exist as a code value.
Paragraph	Case – Paragraph	Can be code value or recno. E.g. Paragraph = "OFL § 11"
AccessGroup	Case – ToAccessGroup	Can be code value or recno. E.g. AccessGroup = "Public"
Notes	Case – Notes	Notes of the case.
ResponsiblePersonIdNumber	Case – OurRef	The external ID of the responsible person. Looks for a contact with ExternalId equal to the property input. It is only possible to set a contactperson in the same enterprise hierarchy as the ADUserContext user as responsible. Provide only ONE of the four possible fields to ID responsible person.
ResponsiblePersonRecno	Case – OurRef	Looks for a contact with Recno equal to the property input.

ResponsiblePersonEmail	Case – OurRef	Looks for a contact with Email equal to the property input.
ResponsiblePersonUserId	Case – OurRef	Looks for a contact with UserId equal to the property input (username, E.g. “no\name.name”)
ResponsibleEnterpriseNumber	Case – ToOrgUnit	<p>The enterprise number of the responsible enterprise.</p> <p>Looks for a contact with ExternalID first, then for one with reference number equal to the property input.</p> <p>If neither this nor the ResponsibleEnterpriseRecno property is set, the responsible enterprise will be set to the responsible person’s enterprise. If the responsible person is not set, the responsible enterprise will be set to the main enterprise of all the users.</p> <p>It is only possible to set the responsible enterprise to an enterprise in the same enterprise hierarchy as the ADUserContext user.</p> <p>Provide only ONE of the two possible fields to ID responsible enterprise.</p>
ResponsibleEnterpriseRecno	Case – ToOrgUnit	This field is a fallback field and is only used if ResponsibleEnterpriseNumber is null or empty. Must be recno of a valid enterprise.
Contacts	Case - ToCaseContact	<p>This adds contacts to the case. Existing case contacts will not be deleted or updated. You can also add an existing contact with a new role.</p> <p>See sub contract CaseContactParameter</p>
ReferringCases	Case – ReferringCases	List of case numbers of referring cases. E.g. (“14/0001”, “14/0002”) Ignored if empty.
Project	Case – Project	Project number of the project this case should be connected to.
ScrapCode	Case – ToScrapCode	Can be code value or recno. E.g. ScrapCode = “B”
Remarks	Case – ToNotes	List of remarks to be added to case

Title (*)	Note – Title	Remark title Mandatory
Content	Note - Notes	Remark content
RemarkType (*)	Note - ToNoteType	Uses code table: Note type Can be code value or recno. Mandatory
Category	Case – ToCaseCategory	Can be code value or recno. E.g. Category = "Leveranse" or "recno:1"
AdditionalFields		Additional fields – see chapter for AdditionalFieldParameter for details, what kind of fields and how to use this parameter.

4.2.2 Output

Data Contract: CaseOperationResult

Output Property	360° field	Comment
Recno	Case – Recno	Unique ID in 360 for the contact
CaseNumber	Case – Name	The case number given in 360.
Successful	-	If true, everything is ok. If false, the operation failed.
ErrorMessage	-	If unsuccessful, this contains a short error message
ErrorDetails	-	If unsuccessful, this contains a full stack trace

4.3 GetCases

Retrieves a list of all cases fulfilling the criteria of all the nonblank properties of the supplied parameter. At least one of the properties (\) must be given. No partial values are allowed.

4.3.1 Input

Data Contract: GetCasesQuery

Input Property	360° field	Comment
ADContextUser	-	Read the general information section.
MaxReturnedCases	-	The maximum number of cases returned by the search. Default is 100.
CaseNumber(\)	Case – Name	The case number of the case you wish to retrieve.
ExternalId(\)		The external id of the case you wish to retrieve. Search uses a combination of Id and Type.
Id	ExternalObject – Id	External id of a case.
Type	ExternalObject - ToExternalType	If not provided, default value “SIF WS Case” will be used. This is the type set via CreateCase if type was not provided when importing a case.
Title(\)	Case – Description	<p>After DLL version 1.0.0.58254: The title search is based on the SQL “LIKE” operator. It is still possible to search for exact matches, but the possibility to search with wildcards is now introduced. For instance, “%” can be used as a substitute for zero or more characters in the title search.</p> <p>Before DLL version: 1.0.0.58254: Only cases matching the exact title will be returned due to the search being based on the SQL “=” operator.</p>
ContactReferenceNumber(\)	Case – ContactConnection	Searches case-contact relations for a private person contact or an enterprise contact with the given property value. Value should be either a PersonalIdNumber or an EnterpriseNumber.
ArchiveCode(\)	Case - ToClassificationCodes	The archive code of the cases you want returned.
ProjectNumber(\)	Case - Project	The project number of the cases you wish to retrieve. E.g. “13-37”

CategoryCode(\)	Case – ToCaseCategory	The case category (code value) to the cases you want to retrieve.
IncludeReferringCases		Boolean value, set true if you want to include case-case-relations in the result. Default value is false.
IncludeReferringDocuments		Boolean value, set true if you want to include case-document-relations in the result. Default value is false.

4.3.2 Output

Data Contract: GetCasesResult

Output Property	360° field	Comment
Cases	Case	A list of CaseResult objects. See the chapter " Data Contract: CaseResult " for details.
Successful	-	Will be true as long as no exception occurred during the execution. This means that successful will be true even if no private persons matching the criteria was found.
ErrorMessage	-	If unsuccessful, this contains a short error message
ErrorDetails	-	If unsuccessful, this contains a full stack trace

4.4 Sub Contracts

This chapter defines data contracts in use by the main data contracts of the Case methods.

4.4.1 Data Contract: CaseContactParameter

Input Property	360° field	Comment
ReferenceNumber(\)	CaseContact – ToContact – Referencenumber	The identification number of the contact. This is PersonalIdNumber for private persons and EnterpriseNumber for enterprises. Use either this field OR the ExternalId-field.
ExternalId(\)	CaseContact – ToContactperson – ExternalID	The identification number of the contact. This is ExternalId for ContactPersons. Use either this field OR the ReferenceNumber-field.
ContactId(\)	CaseContact – ToContact – Referencenumber	This field is obsolete, use ReferenceNumber instead if it is available.
Role(*)	CaseContact – ToRole	The contact's role for the specific case. Can be code or recno. E.g. Role = "Sakspart". The possible roles may be restricted based on the case type.
IsUnofficial		This value can be true, false or null/not included. It is used to define if a contact should be unofficial, which means screened from Public Journal. Should not be set if the access code on the case is U – Unclassified.

4.4.2 Data Contract: ClassCodeParameter

Classification codes have a type and a value. The type refers to the code table *Noark classification type*. Most classification types use codes from the code table *Noark classification code* as values. Some use custom user-supplied values, for instance a personal identification number.

Property	360° field	Comment
Sort(*)	ToClassCodes – Sort	When cases have multiple classification codes, Sort is the index of the code,

		starting at 1. Primary code => 1, secondary => 2, and so on.
ArchiveType(*)	ToClassCodes – ToClassType	The classification type of this particular classification code. Can be code or recno, for instance "FAGKODE PRINSIPP" or «recno:60001»
IsManualText	-	True/false. Set to true if ArchiveCode is a user-supplied value and false if it's a code from <i>Noark classification code</i> .
ArchiveCode	Contact – ToClassCode/Value	The value or classification code itself. If code, it can be either the actual code or its recno, for instance "B32" or "recno:60002".

4.4.3 Data Contract: CaseResult

Property	360° field	Comment
Recno	Document – Recno	Unique id in 360 for the document connected to the case
Title	Case – Description	
CaseNumber	Case – Name	
ExternalId		External ID for the case.
Date	Case – StartDate	The start date of the case.
Status	Case – ToCaseStatus	
ResponsibleEnterprise	Case – ToOrgUnit	Info about the responsible enterprise. See ResponsibleEnterprise
ResponsibleEnterpriseName	Case - ToOrgUnit – SearchName	The name of the responsible enterprise
ResponsiblePerson	Case – OurRef	Info about the responsible person. See ResponsiblePerson
ResponsiblePersonName	Case – OurRef – SearchName	The name of the responsible person
ArchiveCodes	Case – ToClassCodes	A list of all ArchiveType/ArchiveCode pairs on the case.
ReferringCases	Connection: Case - Case	List of case relations.
CaseNumber		Case number of related case
Relation		Code value of relation role
Notes		Notes added to relation
Title		Title of the related case
ReferringDocuments	ActivityCase	List of document relations. Does not include the documents connected to the case as main documents – only documents added as relations.
DocumentNumber	ActivityCase – ToCase	Document number of related

		document
Relation	ActivityCase – ToRole – Code	Code value of relation role
Notes	ActivityCase – Notes	Notes added to relation
Title	ActivityCase – ToCase	Title of the related document
Documents	Document	A list with the meta data for the documents connected to the case. Returns max 100 documents.
DocumentNumber	Document – DocumentNumber	The number of the document connected to the case
Files	File	A list with the meta data for the files connected to the current (active) version of the document. Returns max 100 files.
Recno	File – Recno	Unique id in 360 for the file connected to the document
Title	File – Comment	The title of the file connected to the document
Format	File – Format	The format of the file connected to the document

4.4.4 Data Contract: AdditionalFieldParameter

This can be used for additional 360 fields which are not currently included in the base objects. It can be used for creating and updating fields.

Both standard and custom (customer specific) fields can be set via this parameter, but there are limitations. Only single value string fields can be set. Multi value fields, integer fields etc. can't be set using this parameter. Fields that are connected as a relation to the entity in the meta model, can only be set in some cases, and if so using 360 recno values.

This parameter should only be used after consulting with a Software Innovation employee or someone with knowledge of the 360 meta model.

If the supplied Name does not exist in the customer database, the input is just ignored and no error is thrown.

Property	360° field	Comment
Name(*)	-	The name of the field. This name is the name in the 360 meta model (Not always the same as seen in 360 GUI.) Example: "CF_key"
Value(*)	-	The string value to be set in the field.

5 Document service

Typical URL: [http://\[servername\]:8088/SI.WS.Core/SIF/DocumentService.svc](http://[servername]:8088/SI.WS.Core/SIF/DocumentService.svc)

The fields "document date", "dispatched date" and "journal date" are made available through this service. These are document specific dates that have defined behavior in line with NOARK. That implies the following:

1. Journal date is required to be after both dispatched date and document date.
2. Dispatched date is required to be before journal date but after document date.
3. Document date is required to be before journal date and dispatched date.

5.1 CreateDocument

5.1.1 Input

Data Contract: CreateDocumentParameter

Input Property	360° field	Comment
ADUserContext	-	Read the general information section.
Title(*)	Document - Title	
UnofficialTitle	Document - UnofficialTitle	If not set, Title will be used
DocumentDate	Document - DocumentDate	Defaults to the time the document was received.
Archive	Document – Archive	The Document archive for this document. Can be code or recno. E.g. Archive = "Generelt dokument" Defaults to "Saksdokument" (recno: 2) if not supplied.
Category(*)	Document – ToDocumentCategory	Can be code or recno. E.g. Category = "Referat". View common values.
Status(*)	Document – ToJournalStatus	Can be code or recno. E.g. Status = "J". View common values.
CaseNumber(\)	Document – ToCase	CaseNumber of the case this document belongs to. <i>Not mandatory in all archives.</i> Provide only ONE of the two possible fields to ID case.
CaseExternalId(\)	Document – ToCase	External ID of the case this document belongs to.

		<p>Lookup is based on the combination of Id and Type.</p> <p>Provide only ONE of the two possible fields to ID the case.</p> <p><i>Not mandatory in all archives</i></p>
Id	ExternalObject – Id	External id of the case.
Type	ExternalObject - ToExternalType	If not provided, default value “SIF WS Case” will be used. This is the type set via CreateCase if type was not provided when importing a case.
ResponsiblePersonIdNumber	Document – OurRef - Referencenumber	<p>The external ID of the responsible person.</p> <p>Looks for a contact with ExternalId equal to the property input.</p> <p>It is only possible to set a contactperson in the same enterprise hierarchy as the ADUserContext user as responsible.</p> <p>Provide only ONE of the four possible fields to ID responsible person.</p>
ResponsiblePersonRecno	Document – OurRef - Referencenumber	Looks for a contact with Recno equal to the property input.
ResponsiblePersonEmail	Document – OurRef	Looks for a contact with Email equal to the property input.
ResponsiblePersonUserId	Document – OurRef	Looks for a contact with UserId equal to the property input (username, E.g. “no\name.name”)
ResponsibleEnterpriseNumber	Document – ToOrgUnit	<p>The enterprise number of the responsible enterprise.</p> <p>Looks for a contact with ExternalID first, then for one with reference number equal to the property input.</p> <p>If neither this nor the ResponsibleEnterpriseRecno property is set, the responsible enterprise will be set to the responsible person’s enterprise. If the responsible person is not set, the responsible enterprise will be set to the main enterprise of all the users.</p>

		<p>It is only possible to set the responsible enterprise to an enterprise in the same enterprise hierarchy as the ADUserContext user.</p> <p>Provide only ONE of the two possible fields to ID responsible enterprise.</p>
ResponsibleEnterpriseRecno	Document – ToOrgUnit	This field is a fallback field and is only used if ResponsibleEnterpriseNumber is null or empty. Must be recno of a valid enterprise.
Contacts	Document – ToActivityContact	<p>List of document contacts with associated roles. Must be existing contacts.</p> <p>See the chapter “Data Contract: DocumentContactParameter” for details.</p>
UnregisteredContacts	Document - ToActivityContact	<p>List of unregistered document contacts with associated roles.</p> <p>See the chapter “Data Contract: UnregisteredContactParameter” for details.</p>
SendersReference	Document - ReferenceNo	External reference number.
Access code	Document – Access code	<p>Can be code value or recno. E.g. AccessCode = “U” or “recno:0”</p> <p>Some access code require Paragraph to be specified as well.</p>
Paragraph	Document - ToAuthorization	<p>Can be code or recno. E.g. Paragraph = “OFL § 11”</p> <p>Defaults to the same paragraph as the case.</p>
AccessGroup	Case – ToAccessGroup	Can be code value or recno. E.g. AccessGroup = “Public”
JournalDate	Document - JournalDate	<p>The date the document is closed. Should be used together with a closing status and is ignored otherwise.</p> <p>The following happens if the Status property is set to a closing status:</p>

		<ul style="list-style-type: none"> - If this property is set, the document will gain this date as JournalDate <p>If this property is not set 360 automatically sets the JournalDate to the current date.</p>
DispatchedDate	Document - DispatchedDate	
FiledOnPaper	Document – Paper	
Keywords	Document – Keywords	Adds the given keywords to the case. If they are not found in 360 they are added to the Keyword code table (the user needs Admin rights to create keywords).
Notes	Document – Notes	
ReferringCases	-	List of referring cases, referred by case number. The document-case relation is hardcoded to “reference”. Ignored if empty
ReferringDocuments	-	List of referring documents, referred by document number. Ignored if empty
FilesFromTemplate	-	List of template id’s to create files from. See the chapter “ Data Contract: CreateFileFromTemplateParameter ” for details
Files	-	List of files. See the chapter “ Data Contract: File ” for details
Remarks	Document – ToNotes	List of remarks to be added to case
Title	Note – Title	Remark title Mandatory
Content	Note - Notes	Remark content
RemarkType	Note - ToNoteType	Uses code table: Note type Can be code value or recno. Mandatory
Project	Document – ToProject	The project number of the project you want to connect your document to. E.g. “15-51”
AdditionalFields	Additional fields – see chapter for AdditionalFieldParameter for details, what kind of fields and how to use this parameter.	

5.1.2 Output

Data Contract: DocumentOperationResult

Output Property	360° field	Comment
Recno	Document – Recno	Unique ID in 360 for the document.
DocumentNumber	Document - DocumentNumber	The document number given by 360.
Successful	-	If true, everything is ok. If false, the operation failed.
ErrorMessage	-	If unsuccessful, this contains a short error message
ErrorDetails	-	If unsuccessful, this contains a full stack trace

5.2 UpdateDocument

Updates a document in 360. If a property is set to null, meaning not provided in the parameter passed to the method, the property is not updated. To be able to clear the field in 360 the property has to be set to "". Only works for string properties.

What is possible and not possible with this method:

- It is **not** possible to change or delete files.
- It is **not** possible to add a new version of files.
- It is possible to add files to a document by supplying the Files and DocumentNumber property.

5.2.1 Input

Data Contract: UpdateDocumentParameter

Property	360° field	Comment
ADUserContext	-	Read the general information section.
DocumentNumber(*)	Document – DocumentNumber	Search field. This document will be updated.
DocumentDate	Document - DocumentDate	
Title	Document – Title	
UnofficialTitle	Document – UnofficialTitle	
Category	Document – ToDocumentCategory	Can be code or recno. E.g. Category = "Referat". View

		common values.
Status	Document – ToJournalStatus	Can be code or recno. E.g. Status = “J”. View common values.
ResponsiblePersonIdNumber	Document – OurRef – Referencenumber	<p>The external ID of the responsible person.</p> <p>Looks for a contact with ExternalId equal to the property input.</p> <p>It is only possible to set a contactperson in the same enterprise hierarchy as the ADUserContext user as responsible.</p> <p>Provide only ONE of the four possible fields to ID responsible person.</p>
ResponsiblePersonRecno	Document – OurRef – Referencenumber	Looks for a contact with Recno equal to the property input.
ResponsiblePersonEmail		Looks for a contact with Email equal to the property input.
ResponsiblePersonUserId		Looks for a contact with UserId equal to the property input (username, E.g. “no\name.name”)
ResponsibleEnterpriseNumber	Document – ToOrgUnit	<p>The enterprise number of the responsible enterprise.</p> <p>Looks for a contact with ExternalID first, then for one with reference number equal to the property input.</p> <p>If neither this nor the ResponsibleEnterpriseRecno property is set, the responsible enterprise will be set to the responsible person’s enterprise. If the responsible person is not set, the responsible enterprise will be set to the main enterprise of all the users.</p> <p>It is only possible to set the responsible enterprise to an enterprise in the same enterprise hierarchy as the ADUserContext user.</p>

		Provide only ONE of the two possible fields to ID responsible enterprise.
ResponsibleEnterpriseRecno	Document – ToOrgUnit	This field is a fallback field and is only used if ResponsibleEnterpriseNumber is null or empty. Must be recno of a valid enterprise.
Contacts	Document - ToActivityContact	<p>List of document contacts with associated roles. Must be existing contacts.</p> <p>This adds contacts to the document. Existing document contacts will not be deleted or updated. You can also add an existing contact with a new role.</p> <p>See the chapter “Data Contract: DocumentContactParameter” for details.</p>
UnregisteredContacts	Document - ToActivityContact	<p>List of unregistered document contacts with associated roles.</p> <p>Contacts will be added to the document. Contacts already registered on the document will not be edited or deleted.</p> <p>See the chapter “Data Contract: UnregisteredContactParameter” for details.</p>
SendersReference	Document - ReferenceNo	External reference number.
Access code	Document – Access code	<p>Can be code value or recno. E.g. AccessCode = “U”</p> <p>If the access code requires a paragraph the field paragraph has to offer a valid paragraph.</p> <p>N.B: If a recno value is used, this recno value <u>can not</u> also exist as a code value.</p>
Paragraph	Document - ToAuthorization	<p>Can be code or recno E.g. Paragraph = “OFL § 11”</p> <p>Ignored if used with an access</p>

		code that doesn't require a paragraph.
AccessGroup	Case – ToAccessGroup	Can be code value or recno. E.g. AccessGroup = "Public"
JournalDate	Document- JournalDate	<p>The date the document is closed. Should be used together with a closing status and is ignored otherwise.</p> <p>The following happens if the Status property is set to a closing status:</p> <ul style="list-style-type: none"> - If this property is set, the document will gain this date as JournalDate <p>If this property is not set 360 automatically sets the JournalDate to the current date</p>
DispatchedDate	Document - DispatchedDate	
FiledOnPaper	Document – Paper	
Keywords	Document – Keywords	<p>Updating the keywords on the case by these rules:</p> <ol style="list-style-type: none"> 1) New keywords are added 2) Existing keywords are ignored 3) Missing keywords are ignored
Notes	Document – Notes	
ReferringCases	Document – ToCase	List of referring cases, referred by case number. Ignored if empty.
ReferringDocuments	?	List of referring documents, referred by document number. Ignored if empty.
FilesFromTemplate		List of template id's to create files from. See the chapter " Data Contract: CreateFileFromTemplateParameter " for details
Files	?	List of files that will be added to the document. It is not possible to update previously added files. See the chapter "Data Contract: File" for details.
Remarks	Document – ToNotes	List of remarks to be added to case
Title	Note – Title	Remark title Mandatory
Content	Note - Notes	Remark content

RemarkType	Note - ToNoteType	Uses code table: Note type Can be code value or recno. Mandatory
Project	Document – ToProject	The project number of the project you want to connect your document to. E.g. "15-51"
AdditionalFields		Additional fields – see chapter for AdditionalFieldParameter for details, what kind of fields and how to use this parameter.

5.2.2 Output

Data Contract: DocumentOperationResult

Property	360° field	Comment
Recno	Document – Recno	Unique ID in 360 for the document.
DocumentNumber	Document - DocumentNumber	The document number in 360.
Successful	-	If true, everything is ok. If false, the operation failed.
ErrorMessage	-	If unsuccessful, this contains a short error message
ErrorDetails	-	If unsuccessful, this contains a full stack trace

5.3 SignOffDocument

Method to sign off a document with a response code and an optional note.

5.3.1 Input

Data Contract: SignOffDocumentParameter

Input Property	360° field	Comment
ADContextUser		Read the general information section.
Document(*)	Document – DocumentNumber	The document number of the document to update
ResponseCode(*)	ActivityContact – ResponseCode	The response code. This must be a value from the code table "Response code". E.g. ResponseCode = "TLF"
NoteTitle	Document – ToNotes – Title	This is the description of the note.

		If not supplied, default value is "Merknad ved avskrivning"
Note	Document – ToNotes – Notes	Adds a note to the document, about the sign off. Can be viewed under the tab "Merknader" in Document detail view

5.3.2 Output

Data Contract: [DocumentOperationResult](#)

Same return object as UpdateDocument

5.4 GetDocuments

Take note that this method in addition to getting meta data on documents; may also be used to get file meta data and the physical file(s) inside the document.

5.4.1 Input

Data Contract: GetDocumentsQuery

Input Property	360° field	Comment
ADContextUser		Read the general information section.
MaxReturnedDocuments	-	Maximum number of documents to return. Default 100.
IncludeFileData	-	If the file list should include an output property that contain the files base 64 encoded data. Defaults to false.
CaseNumber(\)	Case – Name	Searches using case number.
CaseExternalId(\)		Searches using external case id, using the combination of Id and Type.
Id	ExternalObject – Id	External id of a case.
Type	ExternalObject - ToExternalType	If not provided, default value “SIF WS Case” will be used. This is the type set via CreateCase if type was not provided when importing a case.
DocumentNumber(\)	Document – DocumentNumber	Searches using document number.
Title(\)	Document – Title	<p>After DLL version 1.0.0.58254: The title search is based on the SQL “LIKE” operator. It is still possible to search for exact matches, but the possibility to search with wildcards is now introduced. For instance, “%” can be used as a substitute for zero or more characters in the title search.</p> <p>Before DLL version 1.0.0.58254: Only documents matching the exact title will be returned due to the search being based on the SQL “=” operator.</p>
ContactReferenceNumber(\)	Document – ToActivityContact – ToContact - Referencenumber	Find documents with a relation to this contact. Value should be either the PersonalIdNumber of a private person or the EnterpriseNumber of an enterprise contact.
ContactExternalId	Document – ToActivityContact – ToContact - ExternalID	Find documents with a relation to this contact, identifies by its ExternalId.

IncludeRemarks		If related remarks should be returned.
IncludeReferringCases		Boolean value, set true if you want to include document-case-relations in the result. Default value is false.
IncludeReferringDocuments		Boolean value, set true if you want to include document-document-relations in the result. Default value is false.

5.4.2 Output

Data Contract: GetDocumentsResult

Output Property	360° field	Comment
Documents	-	List of DocumentResult objects. See the chapter " Data Contract: DocumentResult " for details.
Successful	-	Will be true as long as no exception occurred during the execution. This means that successful will be true even if no private persons matching the criteria was found.
ErrorMessage	-	If unsuccessful, this contains a short error message
ErrorDetails	-	If unsuccessful, this contains a full stack trace

5.5 Sub Contracts

This chapter defines data contracts in use by the main data contracts of the Document methods.

5.5.1 Data Contract: DocumentContactParameter

Input Property	360° field	Comment
ReferenceNumber(\)	ActivityContact – ToContact	The identification number of the contact. This is PersonalIdNumber for private persons and EnterpriseNumber for enterprises. Use either this field OR the ExternalId-field.
ExternalId(\)	ActivityContact – ToContactperson	The identification number of the contact. This is ExternalId for ContactPersons.

		Use either this field OR the ReferenceNumber-field.
Role(*)	ActivityContact – ToRole	The contact's role for the specific document. Can be code or recno. E.g. Role = "Contact". The possible roles may be restricted based on the document type.
IsUnofficial	ActivityContact – Unofficial	This value can be true, false or null/not included. It is used to define if a contact should be unofficial, which means screened from Public Journal. Should not be set if the access code on the document is U – Unclassified.

5.5.2 Data Contract: UnregisteredContactParameter

Used to create an unregistered contact on a document. The fields available in the 360° interface might differ based on settings, or between Service Packs.

Input Property	360° field	Comment
Role(*)	ActivityContact – ToRole	The contact's role for the specific document. Can be code or recno. E.g. Role = "Contact". The possible roles may be restricted based on the document type. View common values.
IsUnofficial	ActivityContact – Unofficial	This value can be true, false or null/not included. It is used to define if a contact should be unofficial, which means screened from Public Journal. Used when the access code on the document is not Unclassified.
ContactName(*)	ActivityContact – Name2 / Activitycontact – Name	Use this to set the name of the unregistered contact. This can be the name of a contact person, a private person or an enterprise. Mandatory field.
ContactCompanyName	ActivityContact – Name	This can be used together with ContactName to set the name of the contact's company.
ReferenceNumber		Reference number
Address		Street address

Country		Country
State		State
ZipCode		ZipCode
ZipPlace		ZipPlace
Email		Email address
MobilePhone		Mobile phone number
Phone		Phone number
Fax		Fax number

5.5.3 Data Contract: ResponsibleEnterprise

Output Property	360° field	Comment
Recno	Contact – Recno	Unique 360 database id field
ReferenceNumber	Contact – Referencenumber	EnterpriseNumber for enterprises.
ExternalId	Contact – ExternalId	ExternalId.
Name	Contact – SearchName	Name.
Email	Contact – E-mail	Email address.

5.5.4 Data Contract: ResponsiblePerson

Output Property	360° field	Comment
Recno	Contact – Recno	Unique 360 database id field
UserId	User – UserID	Username, e.g. "domain\360.admin"
ExternalId	Contact – ExternalId	ExternalId
Name	Contact – SearchName	Name.
Email	Contact – E-mail	Email address.

5.5.5 Data Contract: DocumentResult

Output Property	360° field	Comment
Recno	Document – Recno	
DocumentNumber	Document – DocumentNumber	
CaseNumber	Document – ToCase	
CaseExternalId	Document – ToCase	
Title	Document – Title	
DocumentDate	Document – DocumentDate	
JournalDate	Document – JournalDate	
Category	Document - ToDocumentCategory	Document category

Recno		
Code		
Description		
Type	Document - ToDocumentType	Document type ("I", "U"..). Code table "Journal document type"
Recno		
Code		
Description		
StatusCode	Document – ToJournalStatus – Code	Examples: "J", "R"
StatusDescription	Document – ToJournalStatus – Description	Examples: «Journalført og/eller kontrollert av arkivet», «Reservert av/for saksbehandler»
ResponsibleEnterprise	Document – ToOrgUnit	Info about the responsible enterprise. See ResponsibleEnterprise
ResponsibleEnterpriseName	Document – ToOrgUnit – SearchName	Name on the responsible enterprise.
ResponsiblePerson	Document – OurRef	Info about the responsible person. See ResponsiblePerson
ResponsiblePersonName	Document – OurRef – SearchName	Name on the responsible person.
Contacts	Document	List of DocumentContactResult objects. See the chapter " Data Contract: DocumentContactResult " for details.
ReferringCases	ActivityCase	List of case relations. Does not include the documents main case.
CaseNumber	ActivityCase – ToCase	Case number of related case
Relation	ActivityCase – ToRole – Code	Code value of relation role
Notes	ActivityCase – Notes	Notes added to relation
Title	ActivityCase – ToCase	Title of the related case
ReferringDocuments	Connection: Activity - Activity	List of document relations. Includes documents referring to this document, and documents this document refers to. List is similar to "Document relations" tab in 360 GUI.
DocumentNumber		Document number of related document
Relation		Code value of relation role
Notes		Notes added to relation
Title		Title of the related document
Remarks		List of remarks.
Title		
Content		
TypeCode		

TypeDescription		
CreatedDate		
ModifiedDate		
CreatedBy		
ModifiedBy		
ExpirationDate		
Files		List of files from the current (active) version of the document.
Recno	File - Recno	
Title	File – Comment	
Format	File – Format	
Base64Data		The file data in a base 64 encoded string. Will only be returned if the input property IncludeFileData is set to true.
URL	-	A 360 URL that can be used by authenticated users to view the file. The base of the URL is configured in the SI.WS.Core/web.config key SI.Biz.Serviced.Email.360WebApplicationUrl
Note	File - Note	
Paragraph	File – ToAuthorization	
RelationTypeDescription	File – ToRelationType - Description	
RelationTypeCode	File – ToRelationType - Code	
VersionFormatDescription	File – ToVersionFormat - Description	
VersionFormatCode	File – ToVersionFormat - Code	
Type	File – Type	
ModifiedBy	File – ToFileModifiedBy – ToContact	
CheckedOutBy	File – ToCheckedOutBy – ToContact	
CategoryCode	File – ToFileCategory - Code	
CategoryDescription	File – ToFileCategory - Description	
StatusDescription	File – ToFileStatus - Description	
StatusCode	File – ToFileStatus - Code	

AccessCodeDescription	File – ToAccessCode - Code	
AccessCodeCode	File – ToAccessCode - Description	
DegradeCode	File – ToDegradeCode	Downgrade/declassification code, Norwegian “Avgradering”
DegradeDate	File – DegradeDate	
DisposalCode	File – ToScrapCode	Code value of disposal code, Norwegian “Kassasjonskode”, e.g. “B” or “G”.
DisposalDate	File – ScrapDate	Calculated disposal date, Norwegian “Kassasjonsdato”. This date is in the future and gives info about when the document can be disposed via 360 GUI function for disposal of files.
FiledOnPaper	File – Paper	
PaperLocation	File – PaperLocation	
SignDate	File – ToFileVersion – ToContact	

5.5.6 Data Contract: DocumentContactResult

Output Property	360° field	Comment
ReferenceNumber	Contact - ReferenceNumber	Reference number of the contact – if private person or enterprise.
ExternalId	Contact – ExternalID	External id of the contact – if contact person.
Role	ActivityContact – ToRole	Role of the contact on this document in code value.
SearchName	Contact - SearchName	The name of this contact.

5.5.7 Data contract: CreateFileFromTemplateParameter

This can be used to create new files, either for a new document or to an existing documents. The files will be created based on the template defined. It is not possible to set any other metadata when creating the file from template. Recno to the template can be found in the database or in the web administrator. All available templates can be set.

If more than one file from template is defined, the first one will be added as Main Document and the other ones as Attachements.

If file from template and uploaded files (Se Data Contract: File) are included, the file from template will always be added as Main Document.

Property	360° field	Comment
Title(*)	File - FileName	
TemplateId(*)	File - TemplateID	Recno to the template.

5.5.8 Data Contract: File

There are two ways to upload a file to 360, either as byte array, or by first calling the [FileService.Upload\(\)](#) method, and then providing the file reference it returns in UploadedFileReference.

Property	360° field	Comment
Title(*)	File - FileName	
Format(*)	File - FileFormat	Can be code or recno. E.g. Format = "pdf"
Data(\)		File data
Base64Data(\)		File data as base64 string. Supported, but please use Data instead.
UploadedFileReference(\)	-	This value is returned by FileService.Upload
RelationType	File - ToRelationType	Can be code or recno. E.g. RelationType = "H"
VersionFormat	File - ToVersionFormat	Can be code or recno. E.g. VersionFormat = "A"
Note	File – Note	
Category	File – ToFileCategory	Can be code or recno. E.g. Category = "Brev"
Status	File – ToFileStatus	Can be code or recno. E.g. Status = "B"
AccessCode	File – ToAccessCode	
DegradeCode	File – ToDegradeCode	The "Downgrade" values will only show up in the GUI if both properties DegradeCode and DegradeDate is set.

		Norwegian: "Avgradering".
DegradeDate	File – DegradeDate	The "Downgrade" field will only show up in the GUI if both properties DegradeCode and DegradeDate is set.
FiledOnPaper	File – Paper	
PaperLocation	File – PaperLocation	

6 File service

Typical URL: [http://\[servername\]:8088/SI.WS.Core/SIF/FileService.svc](http://[servername]:8088/SI.WS.Core/SIF/FileService.svc)

6.1 Upload

Streams a file to the 360° server, where it is stored temporarily, and should be referred to in CreateDocument() or UpdateDocument().

6.1.1 Input

Input Property	Comment
FileStream	The file data
User	The uploaded file is stored in this users 360 cache. This user <i>must</i> be the same as the user you authenticate (ADUserContext) with when referring to the file, otherwise it will not be able to locate the it.

6.1.2 Output

Data Contract: UploadFileResult

Output Property	Comment
Successful	If true, the file was uploaded.
FileReference	Use this value to refer to the uploaded file.
ErrorMessage	If unsuccessful, this contains a short error message
ErrorDetails	If unsuccessful, this contains a full stack trace

6.2 CheckOutAndGetFileForExternalControl

Checks out a file for external control, and retrieves the same file as a Stream. The file must later be checked in with the same user and external system, so use this method in combination with UploadAndCheckInFileFromExternalControl

6.2.1 Input

MessageContract: CheckOutAndGetFileForExternalControlMessage

Input Property	Comment
User	Username of the user checking out the file. This locks the file to this user and the same user must check the file back in. The user must also have permission to check out the file.
ExternalSystem	The name of the external system. This can be any string value, but the exact same value must be supplied when checking in the file.
FileRecno	Recno value of the file. This is the file's unique id in the 360° database.

6.2.2 Output

MessageContract: CheckOutAndGetFileForExternalControlResult

Input Property	Comment
FileStream	The file data
CheckOutFileInfo	Information about whether the checkout and retrieval was successful, and meta data about the file if it was successful.
Successful	If true, the file was checked out ok.
ErrorMessage	If unsuccessful, this contains a short error message
ErrorDetails	If unsuccessful, this contains a full stack trace
Title	Title of the file
TemplateId	Recno of the file template
FileRecno	Recno value of the file. This is the file's unique id in the 360° database.

6.3 UploadAndCheckInFileFromExternalControl

Streams a file to the 360° server, where it is checked in by the external system. This can be used on files that have been checked out with the method CheckOutAndGetFileForExternalControl.

6.3.1 Input

MessageContract: UploadAndCheckInFileFromExternalControlMessage

Input Property	Comment
FileStream	The file data

CheckInFileInfo	Information about the file to check in.
FileRecno	Recno value of the file. This is the file's unique id in the 360° database.
Title	Title of the file.
FileFormat	Format of the file, e.g. "docx"
User	Username of the user checking in the file. This user must be the same user that checked out the file.
ExternalSystem	The name of the external system. This can be any string value, but it must be the exact same value that was supplied when checking out the file.

6.3.2 Output

MessageContract: UploadAndCheckInFileFromExternalControlResult

Output Property	Comment
Successful	If true, the file was uploaded and checked in ok.
ErrorMessage	If unsuccessful, this contains a short error message
ErrorDetails	If unsuccessful, this contains a full stack trace
FileRecno	Recno value of the file. This is the file's unique id in the 360° database.

7 Support methods

7.1 Ping

Each web service endpoint has a Ping() method that can be used to test that you have access to the service, that it is correctly installed, and that your user has at least read permissions to 360°. If it returns without throwing an exception, everything is ok.

8 Dependencies to 360

8.1 PDF Converting

Converting of files to PDF (PDF/A) is not done by the web service. This should be done using the 360 Format Converter module which uses a third party application for the conversion. How the conversion is done is best explained with an example:

1. A web service client calls the CreateDocument() or UpdateDocument() with files.
2. The web service creates/updates a document and adds files to it in 360.
3. Part of the 360 Format Converter module is a timer job that copies files to a third party programs TODO folder. Only files that fit preconfigured properties are copied to this folder, a typical property is that the file's document status is "Closed".
4. The third party application converts the files in the TODO folder. The files are then uploaded to 360 to the correct document.
5. The result is a 360 document with the original file(s) and PDF file(s).

The web service has in other words no responsibility of converting files to PDF. The third party application is PixEdit and supports a wide range of file formats for conversion.

9 Web Service client responsibility

A web service client, for example external systems is responsible for the following:

- Creating cases, documents and contacts in the correct order. This means that the web service methods that are available has to be called in the correct logical order. For example you cannot create a document without first creating the case that it belong to.
- Files has to be in the correct format. PDF files is not verified to be PDF/A. No file format validation is done by the web service
- As a general rule, it is client responsibility to validate and clean input to its own needs. Some validation is still done by the 360 business logic, but errors will be in the stack trace format thrown by the business logic.
 - o Take note customer specific validation can be implemented as a customer specific validation component.

10 Common Code Table values

360 has loads of what is referred to as Code Tables and Code Table values. These values are configuration of rows of database values that defines what values can be used as metadata for entities 360. An example is Code Table: Document status, which is a list of all possible document statuses allowed.

Consider the case of document status. Customers with a standard 360 installation usually have the same possible document statuses. However, some do not, and that is why we do not supplement an enum of available values. So any third party that integrates with 360 should verify what values can be used.

When calling a web service method with an input that reference one of these code table values, you can do two things:

- Input "recno:[recno_from_code_table]". Example "recno:6".
- Sometimes only using the recno may work. Example "6". But prefixing it with recno is recommended.
- Input "[code_from_code_table]": Example: "J".

Be warned, valid codes vary by configured web.config value for key SI.Data.BL.LanCode. Since code tables are localized, with different code values for each language, it's important to use codes from the right language. The default language for the SIF web service is Norwegian.

10.1 Case Status

Valid codes vary by configured web.config value for key SI.Data.BL.LanCode.

Recno	Code	Description Norwegian	Description English
4	R	Reservert	Reserved
5	B	Under behandling	In process
6	A	Avsluttet	Closed
7	X	Ikke gjenstand for oppfølging	No follow-up
8	U	Utgår	Cancelled
9	KU	Kopiert utdrag	Copied extract
17	AS	Avsluttet av saksbehandler	Closed by Caseworker

10.2 Document Status

Valid codes vary by configured web.config value for key SI.Data.BL.LanCode.

Recno	Code	Description Norwegian	Description English
1	R	Reservert	Reserved
2	M	Midlertidig journalført	Preliminary recorded
3	S	Registrert av ansvarlig person/leder	Registered by responsible person
4	F	Ferdig fra ansvarlig person/leder	Finished by responsible person

5	E	Ekspedert	Dispatched
6	J	Journalført	Officially recorded
7	A	Registrering avsluttet	Registration completed
8	U	Utgår	Cancelled
9	T	Til godkjenning	In approval process
10	G	Godkjent	Approved
11	I	Ikke godkjent	Not approved
12	K	Til kommentering	For review

10.3 Document Category

Recno	Norwegian	English
110	Dokument inn	Incoming document
111	Dokument ut	Outbound document
112	Referat	Minutes
113	Internt notat med oppfølging	Internal memo with follow-up
114	E-post inn	Incoming e-mail
115	E-post ut	Outbound e-mail
116	Innkommende fax	Incoming faxes
117	Utgående fax	Outbound fax
118	Møteinnkalling	Meeting summons
120	Håndbok	User guide
121	Instruks	Instructions
122	Prosedyre	Procedure
123	Skjema	Form
124	Plan	Plan
125	Kontrakt	Contract
127	Innsynsbegjæringsdokument	Request for access document
216	Kampanjemal	Campaign Template
217	Kampanje eksport	Campaign Export
218	Saksfremlegg/innstilling	Proposition
60005	Internt notat uten oppfølging	Internal memo without follow-up
60006	Vedtak	Decision
95507	Presentasjon	Presentation
95509	Personalhåndbok	Personnell handbook
95514	Rammeavtale	Frame agreement
95515	Support og vedlikeholdsavtale	Support and maintenance agreement
95516	Leieavtale	Rental agreement
95517	Driftsavtale	Operating agreement
95518	Inkøpsavtale	Purchase agreement
95519	Assistanseavtale	Assistance agreement
95520	Oppdragsavtale	Assignment agreement
95524	Intensjonsavtale	Letter of intent

95525	Oppdragsbeskrivelse	Assignment description
--------------	---------------------	------------------------

10.4 Document Contact Role (Activity Contact role)

Recno	Norwegian	English
1	Kontakt	Contact
2	Kunde	Customer
3	Deltaker	Participant
4	Ressurs	Resource
5	Avsender	Sender
6	Mottaker	Recipient
7	Medavsender	Co-sender
8	Kopi til	Carbon copy
9	Godkjenningsinstans	Approval authority
10	Ekstern deltaker	External participant
11	Behandles i	Handled in
20	Blindkopi	Blind Carbon Copy
30	Rådgiver	Advisor
31	Leverandør	Vendor
33	Medforfatter	Co-writer
34	Bestiller	Orderer
35	Datafølgessedel saksparter	Data consignment note case parties
95501	Godkjenner	Approver
95502	Revisjonsansvarlig	Revision responsible