

CRM Plugin for Highrise

Highrise API Documentation:

- <http://developer.37signals.com/highrise/>
- Discussion group -- <http://groups.google.com/group/37signals-api?pli=1>
- Examples of integrations -- <http://highrisehq.com/extras>

Plugin packager URLs (not RESTful)

URL type	URL
Callback URL (called by Ringio after a user authorizes a plugin)	http://{Plugin Domain}/not_rest/create_account?rg_account_id={AccountId}
iframe URL (called by Ringio in the plugin configuration screen)	http://{Plugin Domain}/not_rest/edit_account?rg_account_id_hash={HashAccountId}
Unauthorize URL (called by Ringio after a user unauthorizes a plugin)	http://{Plugin Domain}/not_rest/destroy_account?rg_account_id_hash={HashAccountId}

All URLs will be called with the GET HTTP method.

Replace elements in braces ({}) with the proper parameters:

- {Plugin Domain}: domain of the plugin, for example: testplugin.ringio.com
- {Ringio Account Id}: account id in Ringio, for example: 3

Plugin configuration screen

This GUI will appear under the IFRAME in the Ringio admin page for the plugin.

For the first version, you can only synchronize with one Highrise subdomain.

You can map only one Highrise token per one Ringio user. You can add several maps (token to Ringio user) but there can not be any repeated token or Ringio user (that is, a one-to-one relationship).

[Highrise plugin configuration screen](#)

Tokens and Users
 You can find your token in Highrise under My Info > API Token: Screenshot

Highrise Token	Ringio User	Master User	Actions
89ec1c2a76ea3fdb498b16fe401978	mymail1@exam		Test & Update Delete
	mymail2@exam		Create

Highrise Subdomain https://www..highrisehq.com

As discussed, to avoid possible duplication of information in Ringio after synchronizations or bigger plugin database tables, more complex and time consuming implementation and more resources needed for the plugin server machine, options 2 and 3 will not be offered. Data from users without a Highrise token added will not be synchronized and phone calls from hidden contacts in Ringio will not be synchronized. This shouldn't be a big problem, as small business won't need to add many Highrise tokens and hidden contacts will synchronize all their information to Highrise when they are revealed in Ringio (and both Ringio and Highrise will behave the same way, so it is less confusing for the user).

Highrise data model

In Highrise, you may have People and Companies.

Both People and Companies have contact info such as email or phone.

People belong to Companies but they don't have to.

Companies may have people, but they don't have to.

For EVERY RESOURCE, check attributes owner-id and visible-to: The resources that have visible-to in Highrise are:

- Case - not interesting for Ringio
- Company - it is a contact in Ringio, if the Visibility in Highrise is set to Everyone, then set the Group to Client in Ringio, otherwise, don't (so that the visibility in Ringio is restricted to the owner if the visibility in Highrise is Owner or NamedGroup).

- Deal - not interesting for Ringio
- Email - not interesting for Ringio
- Note
 - For Notes in Ringio: visibility of each note in Ringio is the same as the visibility of its contact, but it is more or less independent from the contact's visibility in Highrise, so we have two options:
 1. Implement independent visibility of notes in Ringio: it is a major change that will require considerable development time.
 2. Restrict to cases where there is no confusion: data that is public both in Ringio and in Highrise, data that is private both in Ringio and in Highrise.

The option chosen is number 2, but giving priority to the contact visibility over the note visibility: in cases when in Highrise there is a public note in a private contact, we will use a (PRIVATE) note in a private contact in Ringio. When in Highrise there is a private note in a public contact, we will use a (PUBLIC) note in a public contact in Ringio. That way, a confused user won't be wondering why the note didn't synchronize, even if it is at the cost of losing some privacy.

- For RingEvents in Ringio: visibility of each RingEvent in Ringio is the same as the visibility of the contact, so we will use that visibility in Highrise. There is no problem, because RingEvent-Notes come from Ringio, so:
 - RingEvent's contact is public in Ringio (Client group) => Highrise visibility is Everyone
 - RingEvent's contact is private in Ringio => Highrise visibility is Owner

- Person - same behaviour as Company.

Management of different server times among Highrise server, the plugin server and Ringio server

As we don't have server time synchronization implemented between the 3 servers, the plugin will need to know the server time from Highrise and from Ringio every time there is a synchronization:

- Ringio will provide their server time in the response of the feed.
- Highrise DOES NOT give their server time as a parameter in their API, so it is necessary to create a fake resource (maybe a Contact) and see which created-at attribute is assigned to it, and then delete that fake resource.

Plugin data model

The plugin will have a table to store the correspondence of ids between resources from Ringio and from Highrise.

It will also need a table that has the configuration data for each user.

How to map users

For each HR user we will accept a token, then the admin will be able to select who should be the owner of those contacts in Ringio.

How to map people and companies in Highrise to Contacts in Ringio

For each user in Highrise, consider the Highrise Persons that they own.

For each Person owned in Highrise, make a Contact in Ringio belonging to the mapped Ringio user.

Highrise field name	Highrise attribute	Ringio field name	Ringio API attribute
first-name	Person.first-name	Name (merged)	Contact.name
last-name	Person.last-name	Name (merged)	Contact.name
title	Person.title	Title	Contact.title
company-id (copy Company Name)	Person.Company.name	Business name	Contact.business
email-address (address)	Person.Contact_data.email_addresses.email_address.address	Email	Contact.data.value Contact.data.type = email
email-address (location)	Person.Contact_data.email_addresses.email_address.location (Work - Home - Other)	Email - label	Contact.data.rel (Work - Home - Other)
phone-number (number)	Person.Contact_data.phone_numbers.phone_number.number	Phone	Contact.data.value Contact.data.type = telephone
phone-	Person.Contact_data.phone_numbers.phone_number.location (Work	Phone - label	Contact.data.rel

number (location)	- Mobile - Fax - Pager - Home - Other)		(Work - Mobile - Fax - Pager - Home - Other)
IM	Person.Contact_data.instant_messengers.instant_messenger.address Person.Contact_data.instant_messengers.instant_messenger.protocol Person.Contact_data.instant_messengers.instant_messenger.location (Work - Personal - Other)	IM	Contact.data.value = address + " in " + protocol Contact.data.type = im Contact.data.rel (Work - Home - Other)
Websites	None	Website -- see note	Contact.data.value Contact.data.type = website Contact.data.rel = OTHER
Twitter	Person.Contact_data.twitter_accounts.twitter_account.username Person.Contact_data.twitter_accounts.twitter_account.location (Personal - Business - Other)	Website (starting with http://twitter.com/username)	Contact.data.value Contact.data.type = website Contact.data.rel = OTHER
Address	Person.Contact_data.addresses.address.{street + city + state + zip + country} Person.Contact_data.addresses.location (Work - Home - Other)	Address	Contact.data.value Contact.data.type = address Contact.data.rel (Work - Home - Other)

Notes:

- Company mapping
- How to populate the Ringio Website field : Make a URL to the Highrise GUI for that Person. E.g. -- <https://ringio.highrisehq.com/people/49510358-richard-hansen> . The subdomain can be obtained by calling /account.xml in the API
- Twitter mapping - Make (a URL with the twitter name)

For each Company in Highrise, make a Contact in Ringio.

Highrise field name	Highrise attribute	Ringio field name	Ringio API attribute
name	Company.name	Business Name	Contact.name
email-address (address)	Company.Contact_data.email_addresses.email_address.address	Email	Contact.data.value Contact.data.type = email
email-address (location)	Company.Contact_data.email_addresses.email_address.location (Work - Home - Other)	Email - label	Contact.data.rel (Work - Home - Other)
phone-number (number)	Company.Contact_data.phone_numbers.phone_number.number	Phone	Contact.data.value Contact.data.type = telephone
phone-number (location)	Company.Contact_data.phone_numbers.phone_number.location (Work - Mobile - Fax - Pager - Home - Other)	Phone - label	Contact.data.rel (Work - Mobile - Fax - Pager - Home - Other)
IM	Person.Contact_data.instant_messengers.instant_messenger.address Person.Contact_data.instant_messengers.instant_messenger.protocol Person.Contact_data.instant_messengers.instant_messenger.location (Work - Personal - Other)	IM	Contact.data.value = address + " in " + protocol Contact.data.type = im Contact.data.rel (Work - Home - Other)
Websites	None	Website -- see note	Contact.data.value Contact.data.type = website Contact.data.rel = OTHER
Twitter	Person.Contact_data.twitter_accounts.username Person.Contact_data.twitter_accounts.location (Personal - Business - Other)	Website (starting with http://twitter.com/username)	Contact.data.value Contact.data.type = website Contact.data.rel = OTHER
Address		Address	

Company.Contact_data.addresses.address.{city + country + state + street + zip}	Company.Contact_data.addresses.location (Work - Home - Other)	Contact.data.value Contact.data.type = address Contact.data.rel (Work - Home - Other)
--	---	---

Notes:

- How to populate the Ringio Website field: Make a URL to the Highrise GUI for that Company. E.g. -- <https://ringio.highrisehq.com/companies/49510189-pinnacle-travel>

How to map notes in Highrise to Notes in Ringio

Every note in Highrise has a subject-id and a subject-type. If the subject type is "Party" then the note belongs to either a (hopefully unambiguous) Person or Company.

Ignore notes with subject type different than "Party".

Since we have figured out the mapping between Highrise People and Highrise Companies to Ringio Contacts, it should be trivial to sync the notes to their corresponding contact in Ringio.

Highrise field name	Highrise attribute	Ringio field name	Ringio API attribute
body	Note.body	text	Note.body
Author	Note.author-id	Owner	Note.author_id
Subject	Note.subject-type = Party Note.subject-name	Name	Note.Contact.name

Highrise notes, more often than not, have escaped HTML formatting, so it would be nice to be able to display the HTML in the note, or alternatively, to strip the HTML. Also, Highrise notes can be very long (whole emails), so we need to truncate the notes to the first x characters, and then embed a link in the note to the HR GUI. (This would mean that we would have to support rendering of URLs as links in Ringio).

As short notes almost won't be affected by stripping HTML tags, and for long notes we will only include the first characters without HTML and a link: in the plugin the HTML tags will always be stripped.

Since we're planning to log Rings as notes, when importing the notes we should skip any note that represents a Ring. (See below)

Mapping Rings to Highrise.

We will map rings to Notes.

- Whenever a Ring is created in Ringio we would create a Note in Highrise for the corresponding contact.
- Notes created in Highrise will not be accepted as new Ring events.
- Ring events are available in Ringio for both contacts taking part in the conversation, so we will create one note in Highrise for each contact affected in Highrise.
- Users can't have notes in Highrise, so Ring events notes will not be created for users, only for contacts.

Highrise field name	Highrise attribute	Ringio field name	Ringio API attribute
body	Note.body	CHECK BELOW THIS TABLE	CHECK BELOW THIS TABLE
Author	Note.Author.id	User	RingEvent.(from/to)contact.user_owner.id
Subject	Note.subject-type = Party Note.subject-name	Name	RingEvent.From.name / RingEvent.To.name

The Highrise Note.body will be filled with this unformatted text in the first version:

RING - DO NOT CHANGE OR DELETE THIS NOTE (this mark must not be removed, or the plugin will think this is a normal note and do all the synchronizations accordingly)
From: RingEvent.from_type + RingEvent.From.name + RingEvent.callerid (which is not a database id, but a caller identifier)
To: RingEvent.to_type + RingEvent.To.name + RingEvent.called_number
Start Time: RingEvent.start_time
Duration: RingEvent.duration
Outcome: RingEvent.outcome
Voicemail: RingEvent.voicemail
Kind: RingEvent.kind (Personal / Team)
Department: RingEvent.department