

# Fluentis AI

**Sergio Buonanno, Rudy Coden**  
Fluentis

# Titolo Corso - Agenda

- Overview
- Model Deployment
- Assistants
- Agent Tools
- Tools
- Agent Objects
- Tool Set
- Mailbox



# Overview



# Overview

**Fluentis AI** costituisce uno dei più recenti sviluppi all'interno dell'ecosistema di **Fluentis**. L'invenzione delle **IA** di tipo *generativo*, non solo rappresenta la terza rivoluzione informatica degli ultimi 50 anni dopo l'introduzione del PC nel 1981 e l'avvento di **Internet** e più specificamente il **World Wide Web** nel 1991, ma, probabilmente, rappresenta la più importante delle tre, con un impatto sulla nostra vita quotidiana e professionale talmente importante da non essere stato ancora completamente compreso. **Fluentis** ha introdotto già nel 2021 una Chat integrata con **Raza**, ma con **Fluentis 2025** si è introdotto un supporto molto più avanzato e integrato. L'aggiunta del supporto alle nuove **IA** di tipo generativo oggi consente a Fluentis di integrare LLM diverse, tra cui, *OpenAI ChatGPT, Google Gemini, DeepSeek, Anthropic, Azure OpenAI*. L'uso di queste LLM può essere fatto in modo esclusivo, oppure utilizzando modelli multipli contemporaneamente, semplicemente aggiungendo specifiche configurazioni all'interno di **ARM**. A seguire andremo ad analizzare alcune delle nuove caratteristiche che l'integrazione con *LLM* di tipo *generativo* consentono di fare, incluse le personalizzazioni per i clienti finali.



# Model Deployment



# Model Deployment

I **Model Deployment** sono i modelli di **IA** disponibili. Fluentis consente di attivare servizi basati su LLM di tipo *generativo* di origine diversa, come: *OpenAI ChatGPT, Google Gemini, DeepSeek, Azure OpenAI, Anthropic Claude*.

Agent object x Model deployment x

Filter

Code

Description

Model deployment [17]

Code	Description	Family	Provider	Purpose	User	Right
agent_azureai_openai_4	Azure OpenAI 4	AzureOpenAI	AzureAI	ChatCompletion	Azure Open AI 4.1	
> agent_azureai_openai_4-mini	Azure OpenAI 4 Mini	AzureOpenAI	AzureAI	ChatCompletion	Azure Open AI 4.1 Mini	
agent_azureai_openai_4-nano	Azure OpenAI 4 Nano	AzureOpenAI	AzureAI	ChatCompletion	Azure Open AI 4.1 Nano	
agent_azureai_openai_5_1	Azure Open AI 5.1	AzureOpenAI	AzureAI	ChatCompletion	Azure OpenAi 5.1	
agent_azureai_openai_5_2	Azure OpenAI 5.2	AzureOpenAI	AzureAI	ChatCompletion	Azure OpenAi 5.2 East US	
agent_azureai_openai_5-mini	Azure Open AI 5 Mini	AzureOpenAI	AzureAI	ChatCompletion	Azure Open AI 5 Mini	
agent_openai_4	Main OpenAI Chat Client	OpenAI	OpenAI	ChatCompletion	OpenAI 4.1	

Property Limit

Key	Value
> key	1fUvKdcnQZqREeYDWtlm10rvt67SFMWfR5Ieq20sub0q73zE1osJQQJ99BLACgEuAYXJ3w3AAAAACOGXQir
baseurl	https://FluentisTrunkAllItaly.cognitiveservices.azure.com/
deployment	gpt-4.1-mini-IT
model	gpt-4.1-mini

# Model Deployment

Le informazioni disponibili per **Model Deployment** sono il **codice**, una **descrizione**, la famiglia cui appartiene il **modello**, che di fatto rappresenta il modello specifico del fornitore, il **provider** che può essere uno tra quelli precedentemente citati e il **Purpose** del modello, cioè, per quale uso è disponibile e tra questi abbiamo:

- **ChatCompletion**: utilizzato nella Chat di Fluentis per inoltrare richieste (prompt) all'IA.
- **Embeddings**: per la generazione di Embeddings utilizzati per ricerche e gli Strumenti degli Agent Objects.
- **ToolsDiscovery**: Ricerca di tool degli Assistant (che vedremo in seguito)
- **AudioEncode**: per la trascrizione di audio in testo
- **AudioDecode**: per lo streaming di testo in audio

Model deployment [17]							
	Code ^	Description	Family	Provider	Purpose	User	Right
	agent_azureai_openai_4	Azure OpenAI 4	AzureOpenAI	AzureAI	ChatCompletion	Azure Open AI 4.1	
>	agent_azureai_openai_4-mini	Azure OpenAI 4 Mini	AzureOpenAI	AzureAI	ChatCompletion	Azure Open AI 4.1 Mini	
	agent_azureai_openai_4-nano	Azure OpenAI 4 Nano	AzureOpenAI	AzureAI	ChatCompletion	Azure Open AI 4.1 Nano	
	agent_azureai_openai_5_1	Azure Open AI 5.1	AzureOpenAI	AzureAI	ChatCompletion	Azure OpenAI 5.1	
	agent_azureai_openai_5_2	Azure OpenAI 5.2	AzureOpenAI	AzureAI	ChatCompletion	Azure OpenAI 5.2 East US	
	agent_azureai_openai_5-mini	Azure Open AI 5 Mini	AzureOpenAI	AzureAI	ChatCompletion	Azure Open AI 5 Mini	
	agent_openai_4	Main OpenAI Chat Client	OpenAI	OpenAI	ChatCompletion	OpenAI 4.1	

# Model Deployment

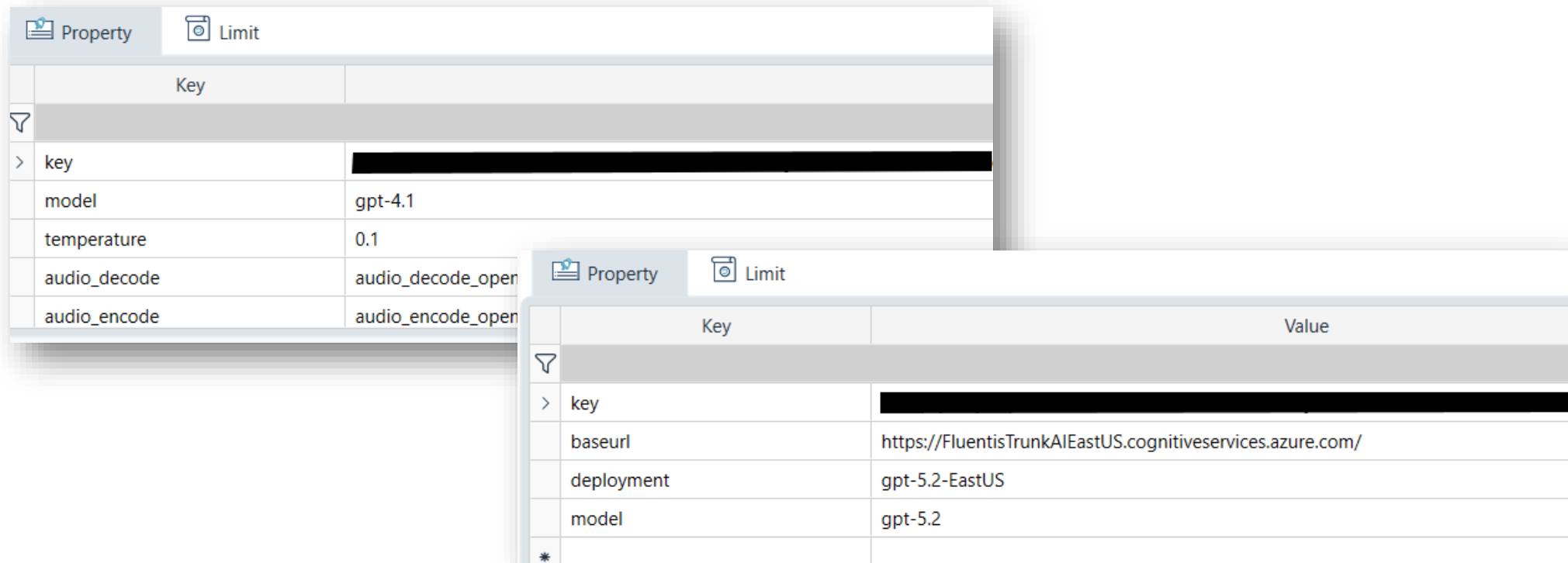
Al *modello* è associato un **utente**, come, per esempio, quello utilizzato in Chat e al quale vengono indirizzate le richieste.

Infine, si possono associare anche dei diritti al fine di limitarne l'uso. L'associazione di diritti è di particolare rilevanza quando si usano gli *Strumenti* che possono accedere a funzionalità riservate di Fluentis.

Purpose	User	Right
ChatCompletion	Azure Open AI 4.1	
ChatCompletion	Azure Open AI 4.1 Mini	
ChatCompletion	Azure Open AI 4.1 Nano	
ChatCompletion	Azure OpenAi 5.1	
ChatCompletion	Azure OpenAi 5.2 East US	
ChatCompletion	Azure Open AI 5 Mini	
ChatCompletion	OpenAi 4.1	

# Model Deployment - Proprietà

A ciascuno modello sono associate delle proprietà necessarie per poterlo utilizzare. Il set di proprietà di base è comune a tutti i modelli mentre alcune altre sono specifiche per ciascuno di essi.



Property		Limit
	Key	
>	key	[REDACTED]
	model	gpt-4.1
	temperature	0.1
	audio_decode	audio_decode_open
	audio_encode	audio_encode_open

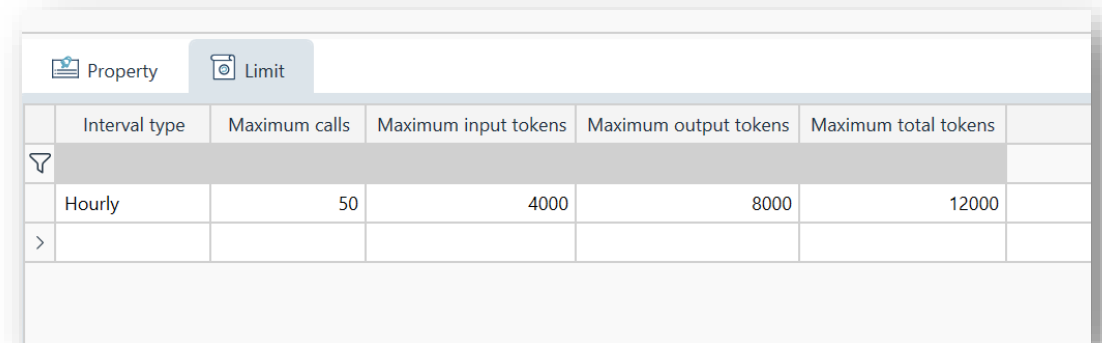
Property		Limit
	Key	Value
>	key	[REDACTED]
	baseurl	<a href="https://FluentisTrunkAIEastUS.cognitiveservices.azure.com/">https://FluentisTrunkAIEastUS.cognitiveservices.azure.com/</a>
	deployment	gpt-5.2-EastUS
	model	gpt-5.2
*		



# Model Deployment - Token limits

Poiché, normalmente, l'uso di una LLM è soggetto ad un costo per token, nel *tab Limits* è possibile specificare dei limiti d'uso del modello che sono i seguenti:

- **Interval type:** l'intervallo temporale a cui i limiti sono applicati. I valori ammessi sono: *orario, giornaliero, mensile*.
- **Maximum calls:** è il numero massimo di chiamate che si possono fare all'LLM nell'intervallo temporale selezionato.
- **Maximum input tokens:** numero massimo di *token* per ogni richiesta inviata all'LLM.
- **Maximum output tokens:** è il numero massimo di *token* restituiti nella risposta della LLM.
- **Maximum total tokens:** massimo numero di *token* nella *Context Window*, cioè input + output *tokens*



	Interval type	Maximum calls	Maximum input tokens	Maximum output tokens	Maximum total tokens
▼	Hourly	50	4000	8000	12000
>					

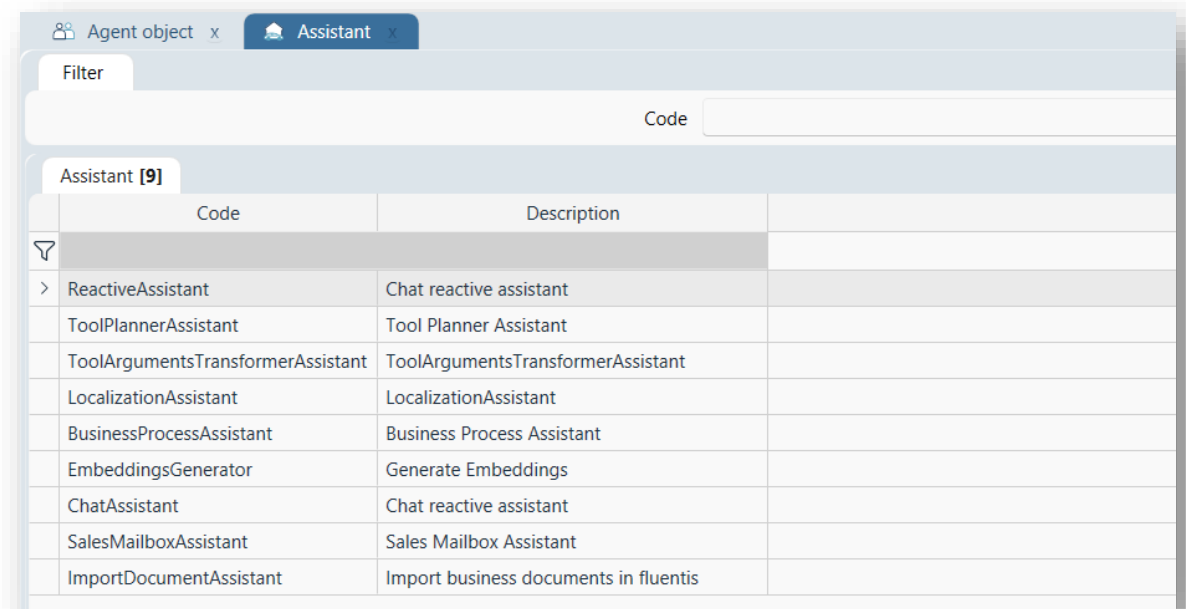


# Assistants



# Assistant

Gli **Assistant** sono oggetti specializzati adibiti all'uso degli Strumenti.  
 Ogni **Assistant** possiede un set di strumenti disponibili configurabili.



The screenshot shows a web application interface with a tab labeled 'Assistant'. Below the tab is a 'Filter' section with a search box labeled 'Code'. Below the filter is a table with 9 rows, each representing an Assistant object. The table has three columns: 'Code', 'Description', and an empty column. The rows are as follows:

Code	Description	
ReactiveAssistant	Chat reactive assistant	
ToolPlannerAssistant	Tool Planner Assistant	
ToolArgumentsTransformerAssistant	ToolArgumentsTransformerAssistant	
LocalizationAssistant	LocalizationAssistant	
BusinessProcessAssistant	Business Process Assistant	
EmbeddingsGenerator	Generate Embeddings	
ChatAssistant	Chat reactive assistant	
SalesMailboxAssistant	Sales Mailbox Assistant	
ImportDocumentAssistant	Import business documents in fluentis	

# Assistant - Activation

Ogni **Assistant** deve essere associato ad un **Model Deployment** così da avere le impostazioni necessarie per poter utilizzare un **LLM** specifico, inoltre, può essere associato ad una società e divisione al fine di contestualizzare il suo utilizzo.

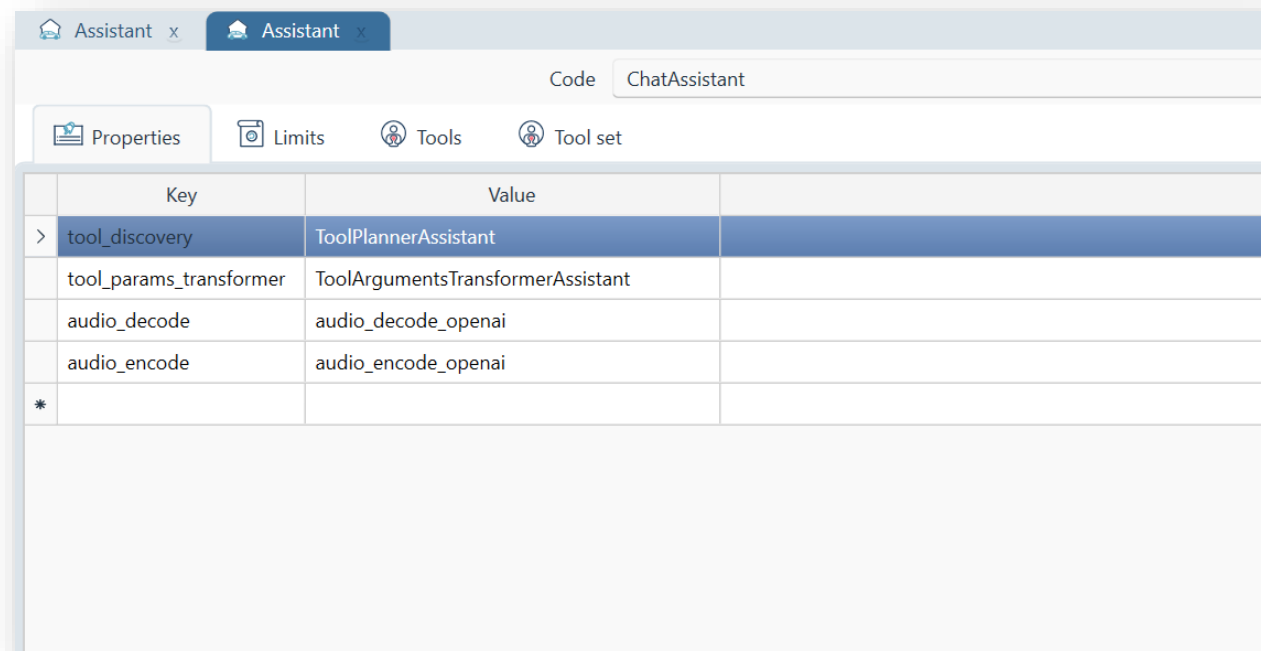
Nel caso non siano specificate l'uso dell'assistente è permesso a tutte le società e divisioni.

Assistant activation [10]

	Assistant	Company	Department	Model deployment
🔍				
>	ReactiveAssistant			agent_azureai_openai_4
	ToolPlannerAssistant			agent_azureai_openai_4
	ToolArgumentsTransformerAssistant			agent_azureai_openai_5_2
	LocalizationAssistant			agent_azureai_openai_5_1
	BusinessProcessAssistant			agent_azureai_openai_4
	EmbeddingsGenerator			embeddings_azureai
	ChatAssistant			agent_azureai_openai_5_2
	SalesMailboxAssistant			agent_azureai_openai_5_1
	ImportDocumentAssistant			agent_azureai_openai_5_2
*				

# Assistant - Properties

Ogni **Assistant** ha a sua volta delle proprietà che dipendono dal modello. Queste proprietà vengono aggiunte nella testata delle richieste inviate al *modello*, insieme alle proprietà che provengono dal **Model Deployment** padre. Ogni proprietà ha una nome o **key** e un **valore** associato.

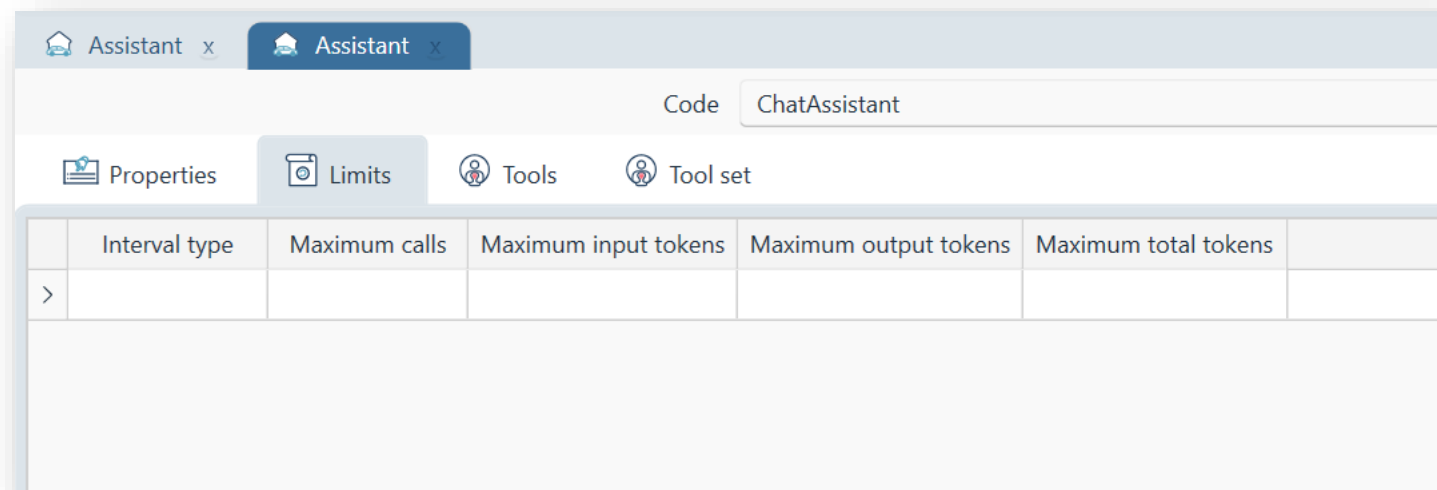


The screenshot shows a web interface for configuring an Assistant. At the top, there are two tabs labeled 'Assistant x'. Below the tabs, there is a 'Code' dropdown menu set to 'ChatAssistant'. A navigation bar contains four items: 'Properties' (selected), 'Limits', 'Tools', and 'Tool set'. Below this is a table with columns 'Key' and 'Value'. The table contains the following entries:

Key	Value
> tool_discovery	ToolPlannerAssistant
tool_params_transformer	ToolArgumentsTransformerAssistant
audio_decode	audio_decode_openai
audio_encode	audio_encode_openai
*	

# Assistant - Limits

Anche per l'**Assistant** è possibile specificare i **limiti**, esattamente come per il Model Deployment padre. I due set di limiti andranno a sovrapporsi e i valori più restrittivi avranno la precedenza, cioè, se **Maximum calls** del Model Deployment padre è maggiore di quello dell'Assistant il valore di quest'ultimo avrà la precedenza.

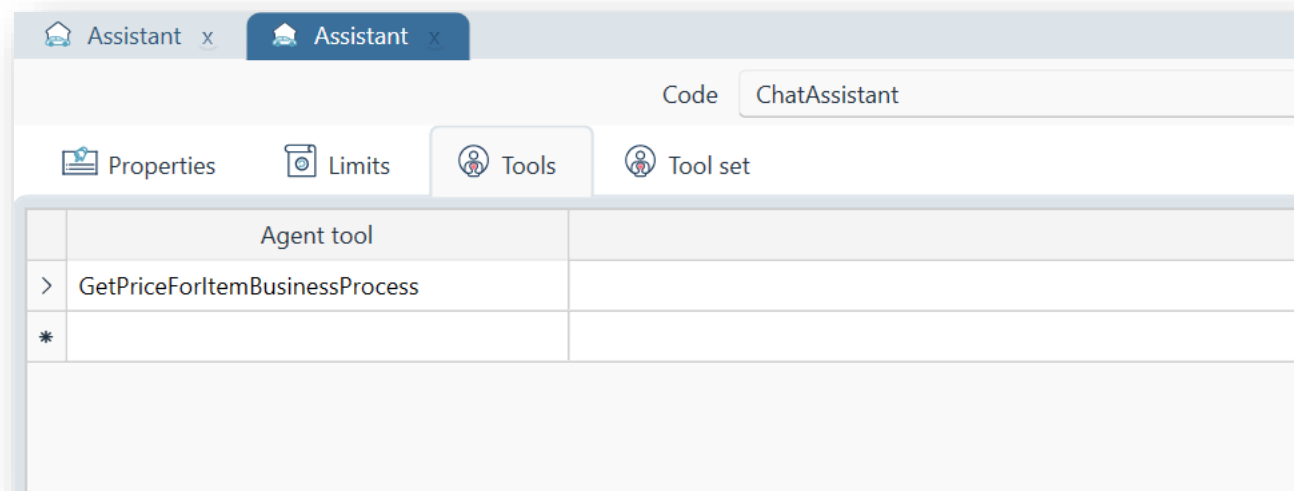


The screenshot shows a web interface for configuring an Assistant. At the top, there are two tabs labeled 'Assistant' with a close button 'x'. Below the tabs, there is a 'Code' field containing 'ChatAssistant'. A navigation bar includes 'Properties', 'Limits' (which is selected), 'Tools', and 'Tool set'. Below the navigation bar is a table with the following structure:

	Interval type	Maximum calls	Maximum input tokens	Maximum output tokens	Maximum total tokens	
>						

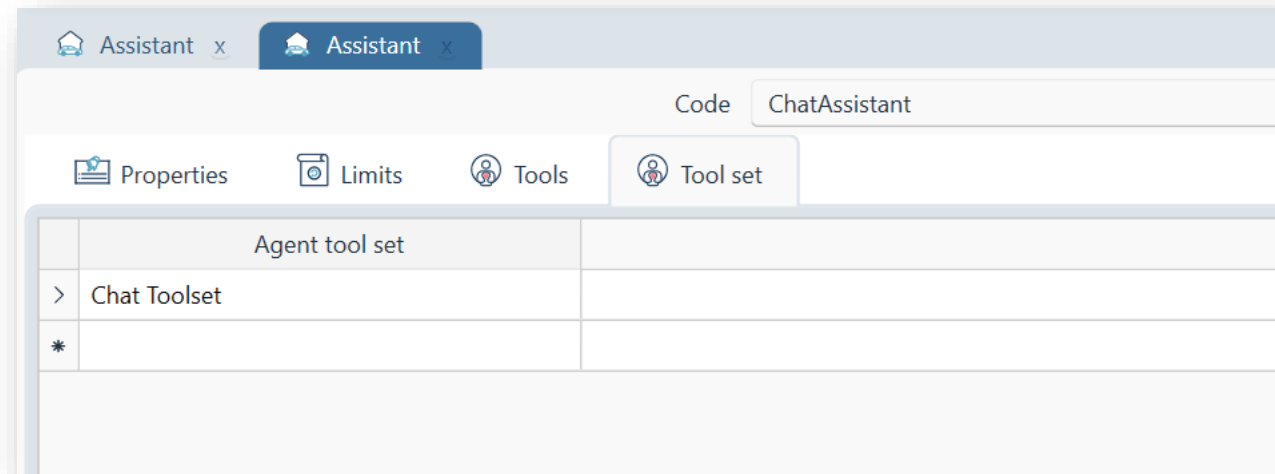
# Assistant - Tools

Agli **Assistant** è associare una lista di **Tools**, un **Tool** rappresenta una funzione o caratteristica richiamabile di Fluentis che andremo ad analizzare in seguito.



# Assistant – Tool-Set

Oltre alla lista di Tools, è possibile assegnare anche un **Tools set**. Un **Tool set** è un gruppo di **Tool** riutilizzabile che può essere assegnato a più assistenti, per esempio, si possono creare **Tool set** per *l'area vendita* o altre aree o moduli di **Fluentis**, che poi possono essere riutilizzati. Invece di specificare i **Tool** di un **Assistant** uno per uno, si può assegnare un **Tool Set** che raggruppa tutti quelli necessari evitando operazioni ridondanti e ripetitive.





# Agent Tools



# Agent Tools

Fluentis mette già a disposizione una lista con numerosi **Agent Tools**; lista che cresce costantemente, ma, soprattutto, che può essere personalizzata. Infatti, i **Tool** che andremo a descrivere più avanti si basano su *funzioni, business process a WebAPI* che non devono essere necessariamente definite a standard ma che possono essere facilmente aggiunti a moduli personalizzati (*custom*), permettendo l'integrazione con **LLM** per espletare funzioni sempre più complesse.

Code	Description
Tool1	test 1
> GetTurnoverForCustomers	calculates the turnover for one or more ledger accounts, given by the supplied filter
GetPriceForItemAndCustomer	Get the preferential price of an item for a given customer (ledger account)
QuerySalesInvoice	tool that queries the db for sales invoices objects, using a personalized filter. the schema of the queried object is (TargetObjectJsonSchema)
QuerySalesOrders	tool that queries the db for sales order objects, using a personalized filter. the schema of the queried object is (TargetObjectJsonSchema)
QueryItems	tool that queries the db for item objects, using a personalized filter. split/separate item class from code when present (consider the usual characters when splitting, like space, slashes, dashes, etc ) the schema of the queried object is (TargetObjectJsonSchema)
QueryLedgerAccounts	tool that queries the db for ledger account (customer) objects, using a personalized filter. the schema of the queried object is (TargetObjectJsonSchema)
VerifyRight	checks if a given user has a certain right
SendMessage	sends a specific message to one user, identified by id
CreateUserTask	Create a user task
GetPriceForItemBusinessProcess	Get the preferential price of an item for a given customer (ledger account)
QueryOwnedBusinessProcesses	fetches a list of all business process instances created for a given user
QueryBusinessProcessStatus	returns the status of a given business process instance
AugmentBusinessProcess	adds additional data to a running business process
CreateSalesOfferForItem	Searches for a preferential price of an item for a given ledger account and creates a new sales offer for the crm contact that requested it
ImportSalesOfferOld	imports a new sales offer using an importData json object
GetObjectSchema	returns the json schema for a known object and a specific format
QueryCrmContact	tool that queries the db for crm contacts, using the provided filter. the schema of the queried object is (TargetObjectJsonSchema)
GetPriceForItemAndCrmContactIds	Get the preferential price of an item for a given crm contact
ValidateJsonObjectAgainstSchema	helper tool that validates the json object against its schema
ExportReportInPdf	Executes the specified report for the specified object and export the result in pdf



# Tools



# Tool

I **Tool** hanno un codice, ma la proprietà più importante è la **descrizione**, infatti, tramite quella l'LLM è in grado di distinguere quale tool è necessario al fine di elaborare una specifica richiesta dell'utente (prompt). L'inserimento di una descrizione completa e precisa è fondamentale per l'utilizzo de Tool.

I tool possono essere abilitati o disabilitati tramite la proprietà **Enabled**.

Il **Tipo** di Tool identifica lo scopo d'uso dello stesso.

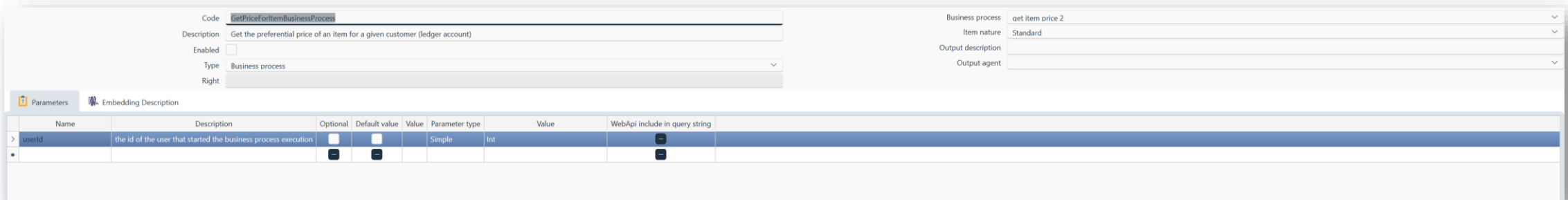
I possibili valori sono i seguenti:

- **Function**: una funzione da richiamare nel *framework* di Fluentis.
- **WebApi**: un End-Point per una chiamata ad una Web Api
- **Search**: funzione di ricerca degli *oggetti business*
- **Business Process**: per l'esecuzione di un **Business Process** di **Fluentis**

Code	GetPriceForItemBusinessProcess	Business process	get item price 2
Description	Get the preferential price of an item for a given customer (ledger account)	Item nature	Standard
Enabled	<input type="checkbox"/>	Output description	
Type	Business process	Output agent	
Right			

# Tool

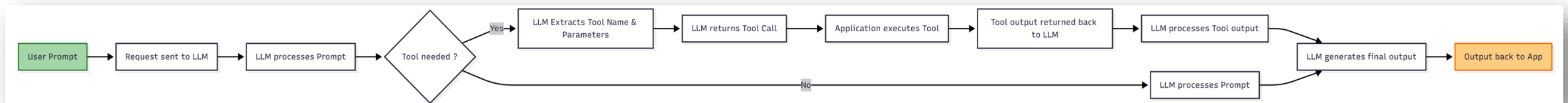
Grazie ai **Tool** il *modello* può far eseguire operazioni che necessitano l'elaborazione da parte di **Fluentis**. Quando si richiede un'operazione specifica al contesto di **Fluentis**, come, per esempio, la giacenza di un articolo, giusto per citare una delle tante operazioni disponibili, l'**LLM** utilizzato non ha modo di sapere la giacenza, ma può dedurre dal contesto di cosa necessita l'utente e verificando le descrizioni disponibili per ciascun **Tool** e i suoi parametri è in grado di eseguire un pattern matching e comunicare, nella risposta a Fluentis, quale **Tool** sia necessario utilizzare; conseguentemente, il *framework* di **Fluentis**, utilizzerà tale informazione per chiamare, al suo interno, la funzione associata, come, per esempio, la ricerca dell'articolo, e poi il calcolo della giacenza e restituirne il risultato all'**LLM** che lo utilizzerà per formulare la risposta finale da presentare all'utente.



Name	Description	Optional	Default value	Value	Parameter type	WebApi include in query string
userid	the id of the user that started the business process execution	<input type="checkbox"/>	<input type="checkbox"/>		Simple Int	<input checked="" type="checkbox"/>

# Tool-Flusso dell'informazione

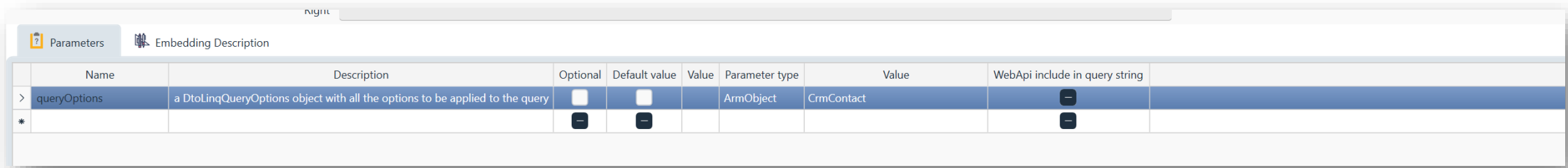
- L'utente invia un *prompt*
- L'applicazione invia il *prompt* al **LLM** tramite richiesta **HTTP**
- **LLM** analizza il prompt e verifica se è necessario usare un **tool**
- Se il **Tool** è necessario
  - **LLM** identifica il **Tool** e i parametri
  - L'applicazione esegue il **Tool**
  - Il risultato del **Tool** viene inviato al **LLM**
  - **LLM** elabora il risultato e genera la risposta finale
- Se il tool non è necessario
  - **LLM** elabora il prompt e genera la risposta finale
  - La risposta finale viene restituita all'utente



# Tool - Parameters

Generalmente tutti i Tool sono di fatto funzioni e come tali richiedono un numero di **parametri** che può essere maggiore o uguale a zero. I **parametri** necessitano di alcune informazioni che vengono passate all'LLM in modo che ne possa estrarre il loro valore dal contesto che poi restituisce a Fluentis per la chiamata alle funzioni interne al *framework*. Le informazioni necessarie sono le seguenti:

- **Description:** necessaria per permettere all'LLM di capire a cosa serve.
- **Optional:** nel caso sia opzionale e l'LLM non ne trovi un valore elaborando il prompt, non verrà restituito un valore corrispondente a Fluentis. In questo caso l'LLM richiederà ulteriori dettagli.
- **Default value:** Nel caso manchi un valore nella richiesta, verrà utilizzato quello assegnato a questa proprietà nella colonna Value.
- **WebApi include in query string:** stabilisce se un parametro verrà aggiunto all'URL della richiesta oppure nel body.

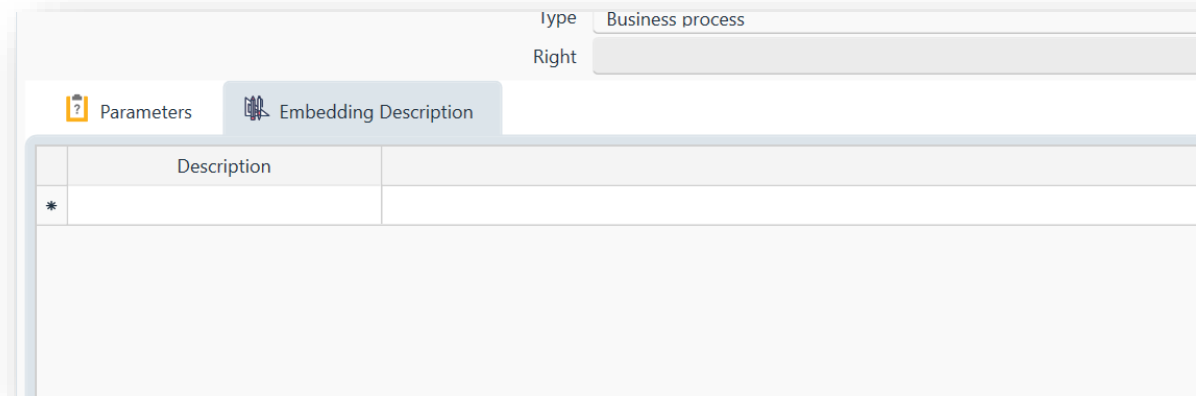


Name	Description	Optional	Default value	Value	Parameter type	Value	WebApi include in query string
> queryOptions	a DtoLinqQueryOptions object with all the options to be applied to the query	<input type="checkbox"/>	<input type="checkbox"/>		ArmObject	CrmContact	[-]
*		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>				[-]

# Tool - Embedding

I Tool possono avere un **Embedding** di descrizione opzionale. Un **embedding** è di fatto un vettore di ricerca. Il valore calcolato dell'**embedding** verrà usato nella ricerca dei Tool da utilizzare, questo per tre ragioni specifiche:

1. Il numero di tool passati all'LLM deve essere ridotto per ridurre la possibilità di errori e i tempi di elaborazione.
  2. Più Tool vengono utilizzati maggiore sarà il numero di token generati e, di conseguenza, il costo per richiesta. Ricordiamo che il numero di *Token* è pari a circa 1,3 volte il numero di parole di un *prompt* ed è sui *token* che viene calcolato il costo di una transazione.
  3. L'**embedding** consente di selezionare un set di tool ridotto per evitare i problemi di cui ai punti 1 e 2.
- La selezione dei *Tool* avviene tramite una specifica funzione di *tool discovery* che genera, da una richiesta, un set di *Tool* che hanno la più alta probabilità di poter fornire le informazioni richieste.



The screenshot shows a configuration window for a tool. At the top right, there are two fields: 'type' with the value 'Business process' and 'Right' which is empty. Below these are two tabs: 'Parameters' (with a question mark icon) and 'Embedding Description' (with a document icon). The 'Embedding Description' tab is active and contains a table with one row. The table has a header row with the column 'Description' and a data row with an asterisk (\*) in the first column and an empty cell in the second column.

type		Business process
Right		
Parameters		
Embedding Description		
	Description	
*		

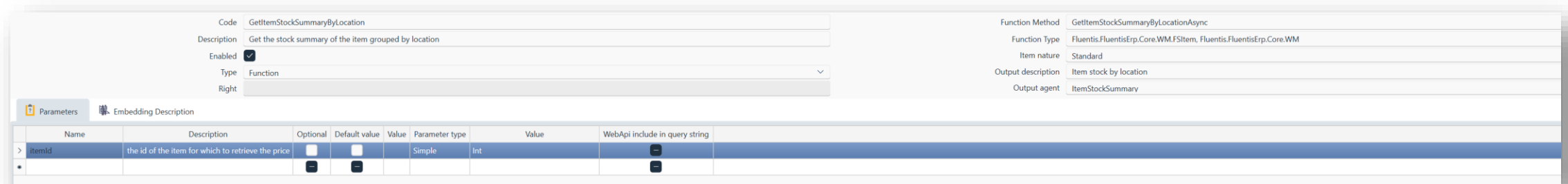
# Tool - Funzioni

I Tool di tipo **funzione** sono utilizzati per richiamare funzionalità specifiche di Fluentis, come, per esempio, il calcolo della giacenza o il fatturato di un cliente e altre ancora.

**Function method** è il nome del metodo statico di Fluentis, mentre **Function type** è la classe e il modulo dove è definita la funzione statica.

Item nature è la stessa *natura* dell'oggetto che abbiamo già visto in molti altri casi all'interno di **Fluentis**.

L'**Output description** è una descrizione sull'oggetto restituito dalla funzione, mentre l'**Agent object** è la proiezione dell'oggetto business che verrà restituito dal **Function Tool**.



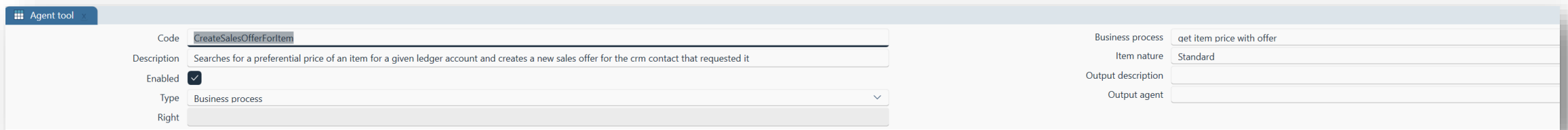
The screenshot shows the configuration interface for a Function Tool. It includes fields for Code, Description, Enabled, Type, Right, Function Method, Function Type, Item nature, Output description, and Output agent. Below these fields is a table for parameters.

Name	Description	Optional	Default value	Value	Parameter type	Value	WebApi include in query string
> itemid	the id of the item for which to retrieve the price	<input type="checkbox"/>	<input type="checkbox"/>		Simple	int	<input type="checkbox"/>
*		<input type="checkbox"/>	<input type="checkbox"/>				<input type="checkbox"/>

# Tool – Business Process

I Tool di tipo **business process** creano ed eseguono un *Business Process*.

Oltre al nome del *business process* indicato dalla proprietà *Business process*, e la natura dell'oggetto di cui abbiamo già discusso, le proprietà: **Output description** e **Output agent** hanno la medesima funzione di quella vista per i **Function Tool**, cioè sono la descrizione e il nome della proiezione, **Agent Object**, restituiti al termine dell'esecuzione del processo.



Code	CreateSalesOfferFortem	Business process	get item price with offer
Description	Searches for a preferential price of an item for a given ledger account and creates a new sales offer for the crm contact that requested it	Item nature	Standard
Enabled	<input checked="" type="checkbox"/>	Output description	
Type	Business process	Output agent	
Right			

# Tool – Search

I Tool di tipo **Search** sono analoghi ai **Function Tool**.

La differenza è che i **Search Tool** servono per i filtri degli oggetti business di **Fluentis** e normalmente vengono utilizzati insieme ad altri Tool per ottenere le informazioni richieste dal cliente, come, per esempio, una lista di clienti e il loro fatturato o similari.

Inizialmente viene utilizzato il **Search Tool** per estrarre la lista dei clienti e poi il Tool per il calcolo del fatturato.



The screenshot shows a configuration form for a tool. The fields are as follows:

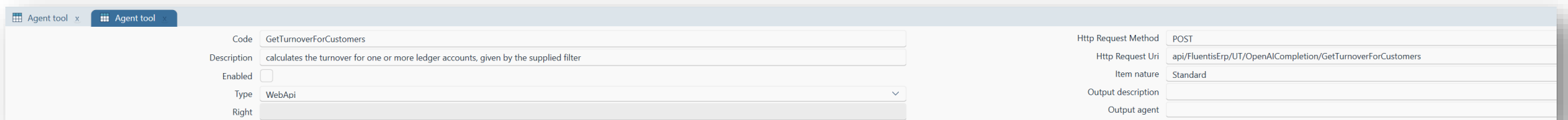
Code	QueryCrmContact	Item nature	Standard
Description	tool that queries the db for crm contacts, using the provided filter. the schema of the queried object is (TargetObject)sonSchema	Output description	
Enabled	<input checked="" type="checkbox"/>	Output agent	
Type	Search		
Right			

# Tool – WebAPI

Anche i Tool di tipo **WebAPI** sono analoghi ai **Function Tool**.

I **WebAPI Tool** necessitano di specificare il tipo di **metodo HTTP** e l'URL del **Web Method**.

Questi vengono utilizzati per chiamare delle **WebAPI** come risposta ad un *prompt*.



The screenshot shows a configuration window for an 'Agent tool'. The tool is named 'GetTurnoverForCustomers' and is of type 'WebApi'. The description is 'calculates the turnover for one or more ledger accounts, given by the supplied filter'. The 'Http Request Method' is set to 'POST' and the 'Http Request Uri' is 'api/FluentisErp/UT/OpenAICompletion/GetTurnoverForCustomers'. The 'Item nature' is 'Standard'. The 'Output description' and 'Output agent' fields are currently empty.

Code	GetTurnoverForCustomers	Http Request Method	POST
Description	calculates the turnover for one or more ledger accounts, given by the supplied filter	Http Request Uri	api/FluentisErp/UT/OpenAICompletion/GetTurnoverForCustomers
Enabled	<input type="checkbox"/>	Item nature	Standard
Type	WebApi	Output description	
Right		Output agent	

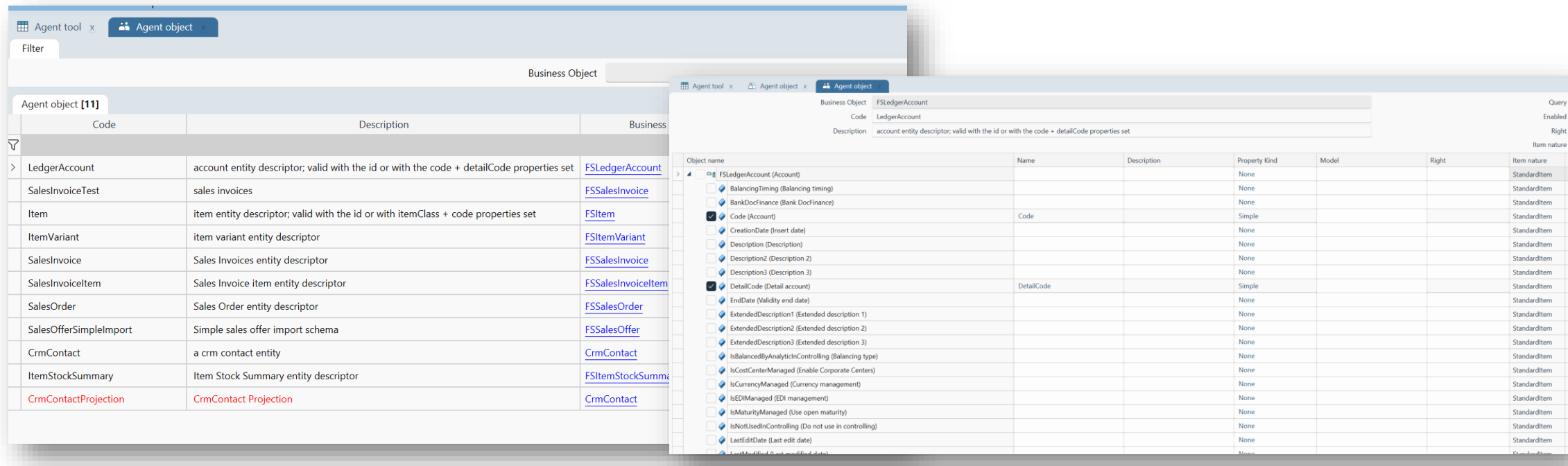


# Agent Objects



# Agent Objects

Gli **Agent Objects** sono le *proiezioni* degli oggetti business di **Fluentis** utilizzate dai **Tool**. Quando viene inviato un *prompt* i *parametri* o i valori restituiti dai tool e i metodi di **Fluentis** possono essere di tipo semplice o di tipo oggetto. L'invio di un *business object* completo sarebbe estremamente oneroso se non praticamente impossibile considerando anche le sue dipendenze, per questo, si creano delle proiezioni degli oggetti che verranno utilizzate come *proxy* degli stessi. Queste *proiezioni* si creano utilizzando strumenti analoghi delle **WebAPI** di **Fluentis**.



The screenshot displays the 'Agent tool' interface with a 'Business Object' tab. It shows a list of agent objects and a detailed view of the 'FSLedgerAccount' object.

Code	Description	Business
LedgerAccount	account entity descriptor; valid with the id or with the code + detailCode properties set	<a href="#">FSLedgerAccount</a>
SalesInvoiceTest	sales invoices	<a href="#">FSSalesInvoice</a>
Item	item entity descriptor; valid with the id or with itemClass + code properties set	<a href="#">FSItem</a>
ItemVariant	item variant entity descriptor	<a href="#">FSItemVariant</a>
SalesInvoice	Sales Invoices entity descriptor	<a href="#">FSSalesInvoice</a>
SalesInvoiceItem	Sales Invoice item entity descriptor	<a href="#">FSSalesInvoiceItem</a>
SalesOrder	Sales Order entity descriptor	<a href="#">FSSalesOrder</a>
SalesOfferSimpleImport	Simple sales offer import schema	<a href="#">FSSalesOffer</a>
CrmContact	a crm contact entity	<a href="#">CrmContact</a>
ItemStockSummary	Item Stock Summary entity descriptor	<a href="#">FSItemStockSummary</a>
<b>CrmContactProjection</b>	<b>CrmContact Projection</b>	<a href="#">CrmContact</a>

Object name	Name	Description	Property Kind	Model	Right	Item nature
FSLedgerAccount (Account)			None			StandardItem
BalancingTiming (Balancing timing)			None			StandardItem
BankDocFinance (Bank DocFinance)			None			StandardItem
<input checked="" type="checkbox"/> Code (Account)	Code		Simple			StandardItem
CreationDate (Insert date)			None			StandardItem
Description (Description)			None			StandardItem
Description2 (Description 2)			None			StandardItem
Description3 (Description 3)			None			StandardItem
<input checked="" type="checkbox"/> DetailCode (Detail account)	DetailCode		Simple			StandardItem
EndDate (Validity end date)			None			StandardItem
ExtendedDescription1 (Extended description 1)			None			StandardItem
ExtendedDescription2 (Extended description 2)			None			StandardItem
ExtendedDescription3 (Extended description 3)			None			StandardItem
IsBalancedByAnalyticControlling (Balancing type)			None			StandardItem
IsCostCenterManaged (Enable Corporate Centers)			None			StandardItem
IsCurrencyManaged (Currency management)			None			StandardItem
IsEDIManaged (EDI management)			None			StandardItem
IsMaturityManaged (Use open maturity)			None			StandardItem
IsNotUsedInControlling (Do not use in controlling)			None			StandardItem
LastModifiedDate (Last edit date)			None			StandardItem
LastModifiedBy (Last modified date)			None			StandardItem

# Agent Objects

Le proprietà di un **Agent Object** sono le seguenti:

- **Code:** il codice utilizzato nelle richieste verso LLM-
- **Description:** una descrizione dell'oggetto e delle sue proprietà. Il *prompting* di queste informazioni va eseguito con particolare attenzione al fine di evitare errori di interpretazione o confusione.
- **Business Object:** l'*oggetto business* per il quale è stata generata la *proiezione*.

Code	Description	Business Object
>		
LedgerAccount	account entity descriptor; valid with the id or with the code + detailCode properties set	<a href="#">FSLedgerAccount</a>
SalesInvoiceTest	sales invoices	<a href="#">FSSalesInvoice</a>
Item	item entity descriptor; valid with the id or with itemClass + code properties set	<a href="#">FSItem</a>
ItemVariant	item variant entity descriptor	<a href="#">FSItemVariant</a>
SalesInvoice	Sales Invoices entity descriptor	<a href="#">FSSalesInvoice</a>
SalesInvoiceItem	Sales Invoice item entity descriptor	<a href="#">FSSalesInvoiceItem</a>
SalesOrder	Sales Order entity descriptor	<a href="#">FSSalesOrder</a>
SalesOfferSimpleImport	Simple sales offer import schema	<a href="#">FSSalesOffer</a>
CrmContact	a crm contact entity	<a href="#">CrmContact</a>
ItemStockSummary	Item Stock Summary entity descriptor	<a href="#">FSItemStockSummary</a>
CrmContactProjection	CrmContact Projection	<a href="#">CrmContact</a>

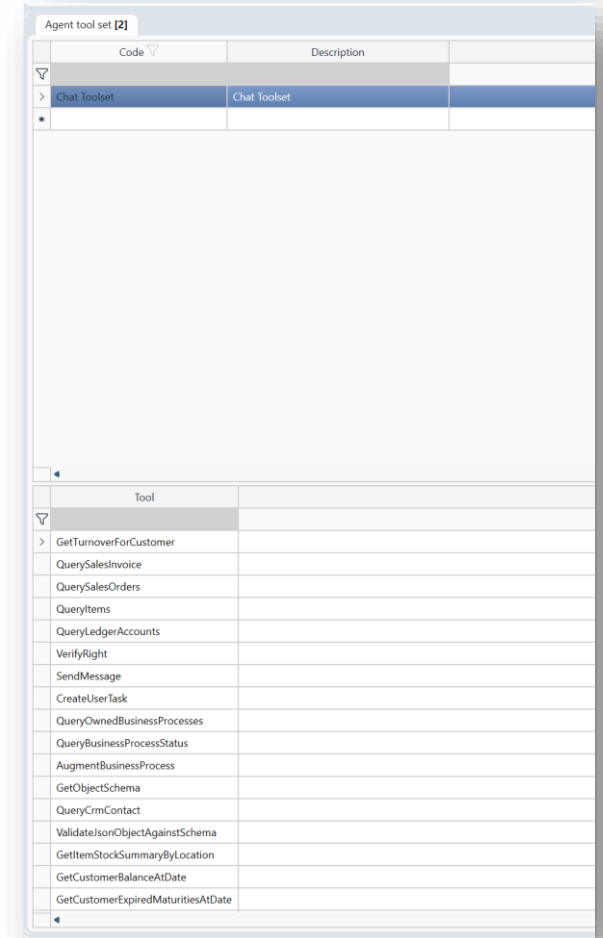


# Agent Tool Set



# Agent Tool set

I **Tools Set** sono un insieme di tool raccolti all'interno di un gruppo. Il vantaggio di avere dei **Tools Set** è quello di poter assegnare un intero set, invece di ogni singolo **Tool**, ad un **Assistant** permettendo di condividere i **Tool** con più **Assistant**. Inoltre, è possibile raggruppare i **Tool** per contesto, come, per esempio: *vendite, acquisti o logistica*.



Code	Description
> Chat Toolset	Chat Toolset
•	
•	
Tool	
> GetTurnoverForCustomer	
QuerySalesInvoice	
QuerySalesOrders	
QueryItems	
QueryLedgerAccounts	
VerifyRight	
SendMessage	
CreateUserTask	
QueryOwnedBusinessProcesses	
QueryBusinessProcessStatus	
AugmentBusinessProcess	
GetObjectSchema	
QueryCrmContact	
ValidateJsonObjectAgainstSchema	
GetItemStockSummaryByLocation	
GetCustomerBalanceAtDate	
GetCustomerExpiredMaturitiesAtDate	



# Mailbox



# Mailbox

Tramite le **Mailbox** è possibile creare degli account di posta elettronica in grado di elaborare task specifici usando l'**IA**. Ciascun indirizzo presidiato utilizzerà un **Assistant** che si occuperà di eseguire il task associato all'indirizzo, come, per esempio, l'invio di un'offerta commerciale per un determinato prodotto. Il destinatario dell'offerta non dovrà fare altro che inviare un'email con il codice prodotto e l'**Assistant** restituirà una email con l'offerta richiesta, questo grazie all'uso di Tool di tipo **Business Process**. Le proprietà di quest'oggetto sono piuttosto semplici, l'unica di esse su cui è importante soffermarsi è l'**Assistant**, che indica, appunto, il codice dell'**Assistant** che processerà il prompt presente nel *body* dell'email.

Mailbox [2]								
	Code	Description	Folder	Email	Last modified date	Last synchronized email...	Server name	Assistant
>	Inbox - fluentis.com	Mailbox for fluentis.com	Inbox	demo.ai@fluentis.com	10/02/2026 16:03	10/02/2026 14:25	webmail.fluentis.com	<a href="#">SalesMailboxAssistant</a>
	TestFolder - fluentis.com	Mailbox for fluentis.com	Inbox/TestFolder	demo.ai@fluentis.com	04/02/2026 00:00	04/02/2026 00:00	webmail.fluentis.com	<a href="#">SalesMailboxAssistant</a>

# Grazie per l'attenzione

Sergio Buonanno, Rudy Coden