Table of Contents

Namespace Uralstech.UAI.Abstraction

Classes

[Message](#)

A message in the conversation.

[Usage](#)

Represents the token usage of an inference request.

Interfaces

[IModelClient](#)

A client for interacting with a model.

Enums

[Role](#)

Role of the creator of a message.

Interface IModelClient

Namespace: Uralstech.UAI.Abstraction

A client for interacting with a model.

```
public interface IModelClient
```

Properties

DefaultModelId

The default model to use for inference.

```
string DefaultModelId { get; set; }
```

Property Value

string

ModelProvider

The provider of the model.

```
string ModelProvider { get; }
```

Property Value

string

Methods

Chat(IReadOnlyList<Message>, IReadOnlyList<Function>, string, int, bool, CancellationToken)

Runs a chat query with the model, with tools.

```
Awaitable<ChatInferenceResult> Chat(IReadOnlyList<Message> messages, IReadOnlyList<Function>
tools, string model = null, int maxToolCalls = 10, bool tryRemoveFilters = false,
CancellationToken token = default)
```

Parameters

messages IReadOnlyList<[Message](#)>

　　The conversation messages.

tools IReadOnlyList<[Function](#)>

　　The tools the model can use.

model string

　　The ID of the model to use.

maxToolCalls int

　　Maximum number of allowed tool calls.

tryRemoveFilters bool

　　Try to remove filters if supported by the model provider.

token CancellationToken

　　Cancellation token.

Returns

Awaitable<[ChatInferenceResult](#)>

　　The model's response.

Chat(IReadOnlyList<Message>, string, bool, CancellationToken)

Runs a chat query with the model.

```
Awaitable<ChatInferenceResult> Chat(IReadOnlyList<Message> messages, string model = null,
bool tryRemoveFilters = false, CancellationToken token = default)
```

Parameters

messages IReadOnlyList<Message>

  The conversation messages.

model string

  The ID of the model to use.

tryRemoveFilters bool

  Try to remove filters if supported by the model provider.

token CancellationToken

  Cancellation token.

Returns

Awaitable<ChatInferenceResult>

  The model's response.

Class Message

Namespace: <u>Uralstech</u>.<u>UAI</u>.<u>Abstraction</u>

A message in the conversation.

```
public record Message
```

Inheritance

object ← Message

Extension Methods

<u>GeminiExtensions.ToGemini(Message)</u> , <u>OAIExtensions.ToOAI(Message)</u>

Constructors

Message(Role, string)

```
public Message(Role role, string content)
```

Parameters

`role` <u>Role</u>

   The role of the creator of the message.

`content` string

   The content of the message.

Fields

Content

The content of the message.

```
public readonly string Content
```

Field Value

string

Role

The role of the creator of the message.

```
public readonly Role Role
```

Field Value

[Role](#)

Methods

ToString()

Returns a string that represents the current object.

```
public override string ToString()
```

Returns

string

A string that represents the current object.

Enum Role

Namespace: [Uralstech](#).[UAI](#).[Abstraction](#)

Role of the creator of a message.

```
public enum Role
```

Extension Methods
[GeminiExtensions.ToGemini(Role)](#) , [OAIExtensions.ToOAI(Role)](#)

Fields

Assistant = 3

AI response.

None = 0

Default value, ignore.

System = 1

System message.

ToolCall = 4

Tool call.

ToolResponse = 5

Tool response.

User = 2

User query.

Class Usage

Namespace: [Uralstech](#).[UAI](#).[Abstraction](#)

Represents the token usage of an inference request.

```
public record Usage
```

Inheritance

object ← Usage

Constructors

Usage(string, int, int)

```
public Usage(string modelProvider, int inputTokens, int outputTokens)
```

Parameters

`modelProvider` string

    The model provider that the usage is associated with.

`inputTokens` int

    The number of input tokens used in the inference request.

`outputTokens` int

    The number of output tokens the inference request generated.

Fields

Empty

Empty value.

```
public static readonly Usage Empty
```

Field Value

[Usage](#)

InputTokens

The number of input tokens used in the inference request.

```
public readonly int InputTokens
```

Field Value

int

ModelProvider

The model provider that the usage is associated with.

```
public readonly string ModelProvider
```

Field Value

string

OutputTokens

The number of output tokens the inference request generated.

```
public readonly int OutputTokens
```

Field Value

int

Methods

ToString()

Returns a string that represents the current object.

```
public override string ToString()
```

Returns

string

A string that represents the current object.

Operators

operator +(Usage, Usage)

Adds two usage objects together.

```
public static Usage operator +(Usage a, Usage b)
```

Parameters

a [Usage](#)

b [Usage](#)

Returns

[Usage](#)

Exceptions

InvalidOperationException

Thrown if the model providers of the objects are different and neither are [Empty](#).

Namespace Uralstech.UAI.Abstraction.Chat

Classes

ChatInferenceResult

Represents the response of a chat request.

Class ChatInferenceResult

Namespace: <u>Uralstech</u>.<u>UAI</u>.<u>Abstraction</u>.<u>Chat</u>

Represents the response of a chat request.

```
public record ChatInferenceResult
```

Inheritance

object ← ChatInferenceResult

Constructors

ChatInferenceResult(Message[], Usage)

```
public ChatInferenceResult(Message[] messages, Usage usage)
```

Parameters

messages <u>Message</u>[]

 The generated content.

usage <u>Usage</u>

 The total token usage of the inference request.

Fields

Messages

The generated content.

```
public readonly Message[] Messages
```

Field Value

[Message](#)[]

Usage

The total token usage of the inference request.

```
public readonly Usage Usage
```

Field Value

[Usage](#)

Methods

ToString()

Returns a string that represents the current object.

```
public override string ToString()
```

Returns

string

A string that represents the current object.

Namespace Uralstech.UAI.Abstraction. Providers.Gemini

Classes

[GeminiExtensions](#)

Extensions to convert between generic and Gemini types.

[GeminiModelClient](#)

Class GeminiExtensions

Namespace: <u>Uralstech</u>.<u>UAI</u>.<u>Abstraction</u>.<u>Providers</u>.<u>Gemini</u>

Extensions to convert between generic and Gemini types.

```
public static class GeminiExtensions
```

Inheritance

object ← GeminiExtensions

Methods

ToGemini(IReadOnlyList<Message>, out GeminiContent)

Converts a collection of generic messages to an array of Gemini messages.

```
public static UGemini.Models.Content.GeminiContent[] ToGemini(this IReadOnlyList<Message>
messages, out UGemini.Models.Content.GeminiContent systemMessage)
```

Parameters

`messages` IReadOnlyList<<u>Message</u>>

`systemMessage` GeminiContent

   Special handling for system role.

Returns

GeminiContent[]

Exceptions

InvalidOperationException

   Thrown if more than one system message is found in the messages.

ToGemini(IReadOnlyList<Function>)

Converts a collection of generic functions to an array of UGemini.Models.Generation.Tools.Declaration.GeminiTools.

```
public static UGemini.Models.Generation.Tools.Declaration.GeminiTool ToGemini(this
IReadOnlyList<Function> functions)
```

Parameters

`functions` IReadOnlyList<[Function](#)>

Returns

GeminiTool

ToGemini(Message)

Converts a generic message to an Gemini message. [System](#) will throw an error as it requires special handling.

```
public static UGemini.Models.Content.GeminiContent ToGemini(this Message message)
```

Parameters

`message` [Message](#)

Returns

GeminiContent

ToGemini(Role)

Converts a generic role to an Gemini role. [System](#) will throw an error as it requires special handling.

```
public static UGemini.Models.Content.GeminiRole ToGemini(this Role role)
```

Parameters

`role` [Role](#)

Returns

GeminiRole

Exceptions

NotImplementedException

   Thrown if a generic role has no known Gemini equivalent.

ToGemini(Function)

Converts a generic function to an Gemini function.

```
public static UGemini.Models.Generation.Tools.Declaration.GeminiFunctionDeclaration
ToGemini(this Function function)
```

Parameters

`function` [Function](#)

Returns

GeminiFunctionDeclaration

Exceptions

NotImplementedException

   Thrown if a generic [ParameterType](#) has no known Gemini equivalent.

ToGeneric(IReadOnlyList<GeminiContent>)

Converts a collection of Gemini messages to an array of generic messages.

```
public static Message[] ToGeneric(this IReadOnlyList<UGemini.Models.Content.GeminiContent>
messages)
```

Parameters

`messages` IReadOnlyList<UGemini.Models.Content.GeminiContent>

Returns

[Message](#)[]

ToGeneric(GeminiContent)

Converts an Gemini message to a generic message.

```
public static Message ToGeneric(this UGemini.Models.Content.GeminiContent message)
```

Parameters

`message` GeminiContent

Returns

[Message](#)

ToGeneric(GeminiRole)

Converts an Gemini role to a generic role.

```
public static Role ToGeneric(this UGemini.Models.Content.GeminiRole role)
```

Parameters

`role` GeminiRole

Returns

[Role](#)

Exceptions

NotImplementedException

    Thrown if a Gemini role has no known generic equivalent.

ToGeneric(GeminiUsageMetadata)

Converts an Gemini function to a generic function.

```
public static Usage ToGeneric(this UGemini.Models.Generation.Candidate.GeminiUsageMetadata
usage)
```

Parameters

usage GeminiUsageMetadata

Returns

[Usage](#)

Class GeminiModelClient

Namespace: Uralstech.UAI.Abstraction.Providers.Gemini

```
public class GeminiModelClient : IModelClient
```

Inheritance

object ← GeminiModelClient

Implements

IModelClient

Fields

ModelProvider

Constant for the model provider.

```
public const string ModelProvider = "gemini"
```

Field Value

string

Properties

DefaultModelId

The default model to use for inference.

```
public string DefaultModelId { get; set; }
```

Property Value

string

Methods

Chat(IReadOnlyList<Message>, IReadOnlyList<Function>, string, int, bool, CancellationToken)

Runs a chat query with the model, with tools.

```
public Awaitable<ChatInferenceResult> Chat(IReadOnlyList<Message> messages,
IReadOnlyList<Function> tools, string model = null, int maxToolCalls = 10, bool
tryRemoveFilters = false, CancellationToken token = default)
```

Parameters

`messages` IReadOnlyList<[Message](#)>

    The conversation messages.

`tools` IReadOnlyList<[Function](#)>

    The tools the model can use.

`model` string

    The ID of the model to use.

`maxToolCalls` int

    Maximum number of allowed tool calls.

`tryRemoveFilters` bool

    Try to remove filters if supported by the model provider.

`token` CancellationToken

    Cancellation token.

Returns

Awaitable<[ChatInferenceResult](#)>

    The model's response.

Chat(IReadOnlyList<Message>, string, bool, CancellationToken)

Runs a chat query with the model.

```
public Awaitable<ChatInferenceResult> Chat(IReadOnlyList<Message> messages, string model =
null, bool tryRemoveFilters = false, CancellationToken token = default)
```

Parameters

`messages` IReadOnlyList<Message>

   The conversation messages.

`model` string

   The ID of the model to use.

`tryRemoveFilters` bool

   Try to remove filters if supported by the model provider.

`token` CancellationToken

   Cancellation token.

Returns

Awaitable<ChatInferenceResult>

   The model's response.

Namespace Uralstech.UAI.Abstraction.Providers.OAI

Classes

[OAIExtensions](#)

Extensions to convert between generic and OpenAI types.

[OAIModelClient](#)

A client for interacting with OpenAI models through [IModelClient](#).

Class OAIExtensions

Namespace: [Uralstech](#).[UAI](#).[Abstraction](#).[Providers](#).[OAI](#)

Extensions to convert between generic and OpenAI types.

```
public static class OAIExtensions
```

Inheritance

object ← OAIExtensions

Methods

ToGeneric(Message)

Converts an OpenAI message to a generic message.

```
public static Message ToGeneric(this OpenAI.Chat.Message message)
```

Parameters

`message` Message

Returns

[Message](#)

ToGeneric(Role)

Converts an OpenAI role to a generic role.

```
public static Role ToGeneric(this OpenAI.Role role)
```

Parameters

`role` Role

Returns

[Role](#)

Exceptions

NotImplementedException

   Thrown if a OpenAI role has no known generic equivalent.

ToGeneric(Usage)

Converts an OpenAI function to a generic function.

```
public static Usage ToGeneric(this OpenAI.Usage usage)
```

Parameters

`usage` Usage

Returns

[Usage](#)

ToGeneric(IReadOnlyList<Message>)

Converts a collection of OpenAI messages to an array of generic messages.

```
public static Message[] ToGeneric(this IReadOnlyList<OpenAI.Chat.Message> messages)
```

Parameters

`messages` IReadOnlyList<OpenAI.Chat.Message>

Returns

[Message](#)[]

ToOAI(IReadOnlyList<Message>)

Converts a collection of generic messages to an array of OpenAI messages.

```
public static OpenAI.Chat.Message[] ToOAI(this IReadOnlyList<Message> messages)
```

Parameters

`messages` IReadOnlyList<Message>

Returns

Message[]

ToOAI(IReadOnlyList<Function>)

Converts a collection of generic functions to an array of OpenAI.Tools.

```
public static OpenAI.Tool[] ToOAI(this IReadOnlyList<Function> functions)
```

Parameters

`functions` IReadOnlyList<Function>

Returns

Tool[]

ToOAI(Message)

Converts a generic message to an OpenAI message.

```
public static OpenAI.Chat.Message ToOAI(this Message message)
```

Parameters

`message` Message

Returns

Message

ToOAI(Role)

Converts a generic role to an OpenAI role.

```
public static OpenAI.Role ToOAI(this Role role)
```

Parameters

role Role

Returns

Role

Exceptions

NotImplementedException

Thrown if a generic role has no known OpenAI equivalent.

ToOAI(Function)

Converts a generic function to an OpenAI function.

```
public static OpenAI.Function ToOAI(this Function function)
```

Parameters

function Function

Returns

Function

Exceptions

NotImplementedException

Thrown if a generic ParameterType has no known OpenAI equivalent.

Class OAIModelClient

Namespace: <u>Uralstech</u>.<u>UAI</u>.<u>Abstraction</u>.<u>Providers</u>.<u>OAI</u>

A client for interacting with OpenAI models through <u>IModelClient</u>.

```
public class OAIModelClient : IModelClient
```

Inheritance

object ← OAIModelClient

Implements

<u>IModelClient</u>

Fields

ModelProvider

Constant for the model provider.

```
public const string ModelProvider = "openai"
```

Field Value

string

Properties

DefaultModelId

The default model to use for inference.

```
public string DefaultModelId { get; set; }
```

Property Value

string

Methods

Chat(IReadOnlyList<Message>, IReadOnlyList<Function>, string, int, bool, CancellationToken)

Runs a chat query with the model, with tools.

```
public Awaitable<ChatInferenceResult> Chat(IReadOnlyList<Message> messages,
IReadOnlyList<Function> tools, string model = null, int maxToolCalls = 10, bool
tryRemoveFilters = false, CancellationToken token = default)
```

Parameters

`messages` IReadOnlyList<[Message](#)>

    The conversation messages.

`tools` IReadOnlyList<[Function](#)>

    The tools the model can use.

`model` string

    The ID of the model to use.

`maxToolCalls` int

    Maximum number of allowed tool calls.

`tryRemoveFilters` bool

    Try to remove filters if supported by the model provider.

`token` CancellationToken

    Cancellation token.

Returns

Awaitable<[ChatInferenceResult](#)>

The model's response.

Chat(IReadOnlyList<Message>, string, bool, CancellationToken)

Runs a chat query with the model.

```
public Awaitable<ChatInferenceResult> Chat(IReadOnlyList<Message> messages, string model =
null, bool tryRemoveFilters = false, CancellationToken token = default)
```

Parameters

`messages` IReadOnlyList<Message>

The conversation messages.

`model` string

The ID of the model to use.

`tryRemoveFilters` bool

Try to remove filters if supported by the model provider.

`token` CancellationToken

Cancellation token.

Returns

Awaitable<ChatInferenceResult>

The model's response.

Namespace Uralstech.UAI.Abstraction.Tools

Classes

Function

A function that can be invoked by a model.

Parameter

Declares a parameter for a Function.

Enums

ParameterType

Type of the parameter.

Class Function

Namespace:

A function that can be invoked by a model.

```
public class Function
```

Inheritance

object ← Function

Extension Methods

GeminiExtensions.ToGemini(Function) , OAIExtensions.ToOAI(Function)

Constructors

Function(string, string, Parameter[], Func<JToken, Awaitable<JObject>>)

```
public Function(string name, string description, Parameter[] parameters = null, Func<JToken,
Awaitable<JObject>> onInvokeAsync = null)
```

Parameters

`name` string

   The name/ID of the function.

`description` string

   The description of the function.

`parameters` Parameter[]

   The parameters of the function.

`onInvokeAsync` Func<JToken, Awaitable<JObject>>

   The underlying C# function that will be invoked.

Fields

Description

The description of the function.

```
public readonly string Description
```

Field Value

string

Name

The name/ID of the function.

```
public readonly string Name
```

Field Value

string

OnInvokeAsync

The underlying C# function that will be invoked.

```
public Func<JToken, Awaitable<JObject>> OnInvokeAsync
```

Field Value

Func<JToken, Awaitable<JObject>>

Parameters

The parameters of the function.

```
public readonly Parameter[] Parameters
```

Field Value

[Parameter](#)[]

Methods

ToString()

Returns a string that represents the current object.

```
public override string ToString()
```

Returns

string

A string that represents the current object.

Class Parameter

Namespace: [Uralstech](#).[UAI](#).[Abstraction](#).[Tools](#)

Declares a parameter for a [Function](#).

```
public record Parameter
```

Inheritance

object ← Parameter

Fields

Description

The description of the parameter.

```
public string Description
```

Field Value

string

Enum

Enum values for the parameter, if the type is [String](#).

```
public string[] Enum
```

Field Value

string[]

Name

The name of the parameter.

```
public string Name
```

Field Value

string

Required

Is the parameter required?

```
public bool Required
```

Field Value

bool

Type

The type of the parameter.

```
public ParameterType Type
```

Field Value

ParameterType

Methods

ToString()

Returns a string that represents the current object.

```
public override string ToString()
```

Returns

string

A string that represents the current object.

Enum ParameterType

Namespace: [Uralstech](#).[UAI](#).[Abstraction](#).[Tools](#)

Type of the parameter.

```
public enum ParameterType
```

Fields

Boolean = 3

Boolean type.

Float = 1

Float type.

Integer = 0

Integer type.

String = 2

String type.