

DYALOC

Glasgow 2024

New Tatin Packages



Brian Becker



What is a Tatin?



A tasty vegan apple tarte tatin
(You're welcome Mark!)



The logo for Tatin, a package manager for
tastefully packaged APL code

Quick Survey

- How many have used a Tatin package?
- How many have written a Tatin package?
- How many have some code that might make a good Tatin Package?

https://tatin.dev



List of packages

Package name	Description	Major Versions	Project URL	OS	UC	Tags
aplteam-ADOC	Automated generation of documentation	1	github.com	Lin, Mac, Win	Yes	documentation
aplteam-APLGit2	Git interface from Dyalog APL via Git Bash	1	github.com	Lin, Mac, Win	Yes	apl-git-interface
aplteam-APLGUI	Collection of GUI utilities	1	github.com	Win		gui-tools,gui
aplteam-APLProcess	Start an APL process from within Dyalog APL	1	github.com	Lin, Mac, Win		process
aplteam-APLTreeUtils2	General utilities required by most members of the APLTree library	1	github.com	Lin, Mac, Win		tools,utilities
aplteam-Cider	A project manager for Dyalog APL that cooperates with Tatin	1	github.com	Lin, Mac, Win	Yes	project-management
aplteam-				Lin,		

Current Tatin Packages

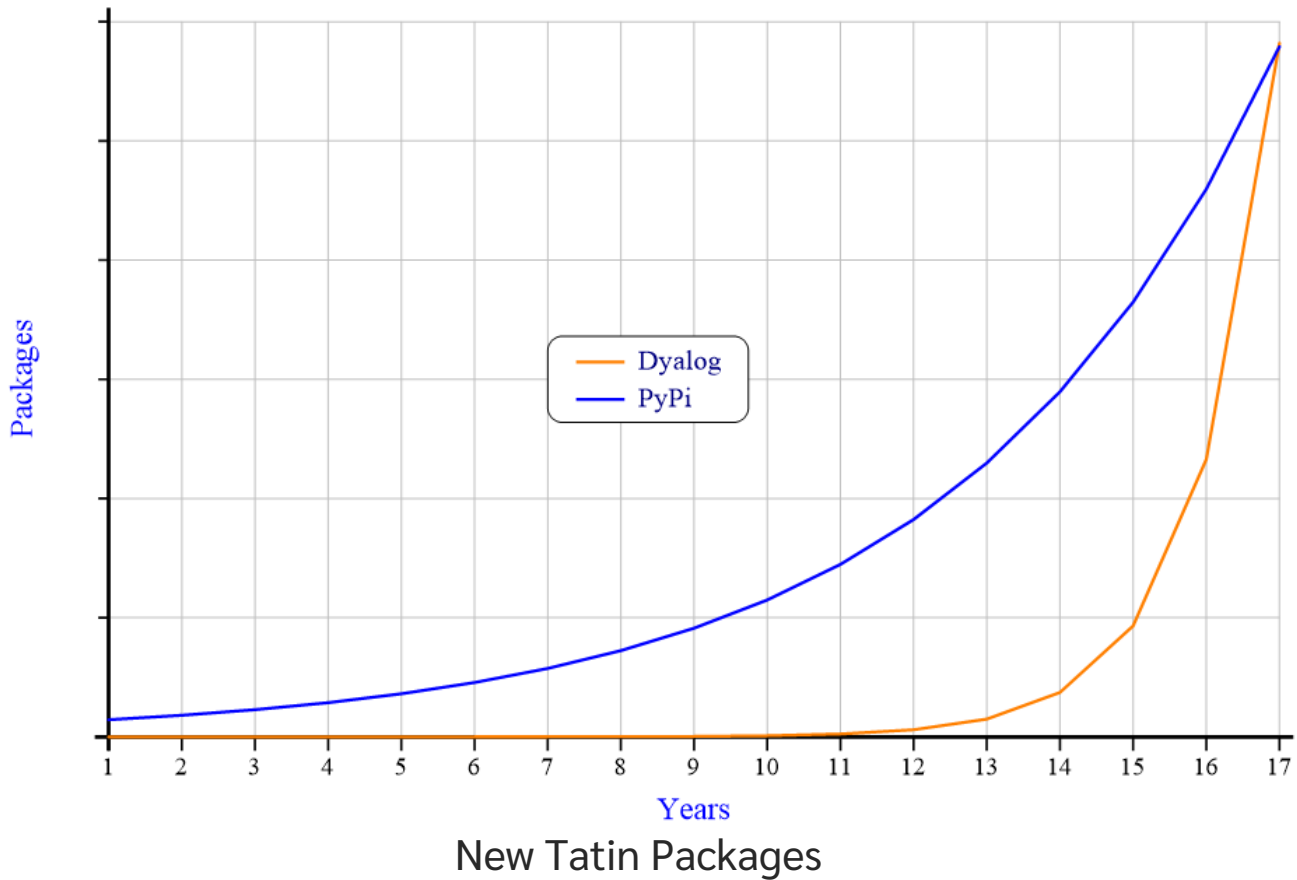
Packager	Number of Packages
APL Team	44
Davin Church	4
Dyalog	5 (3 new in 2024)

Thank you Kai!

- ◆ For sharing quality utility libraries for decades.
- ◆ For developing and hosting Tatin
- ◆ For everything else you've done for the APL community

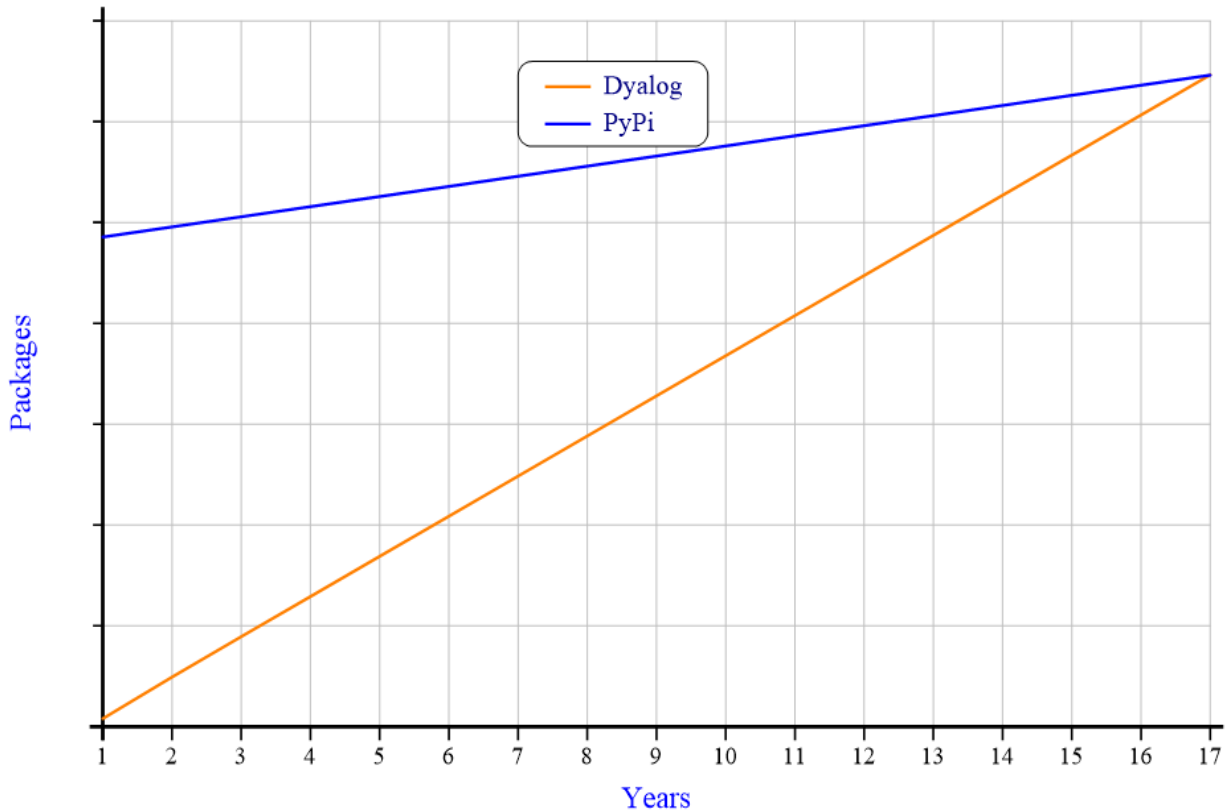
At 150% Growth... assuming same growth as 2023-2024

Tatin v PyPi



Tatin v PyPi

assuming same growth as 2023-2024



New Dyalog Packages

- ◆ NuGet – makes it easier to use .NET packages from APL
- ◆ APLProcess – spawn a new APL interpreter in another process
- ◆ OpenAI – Implements an interface to the OpenAI API

NuGet

- ◆ A tool to help use NuGet packages from Dyalog APL.

NuGet

The image is a screenshot of the NuGet website homepage. At the top, there is a navigation bar with the NuGet logo and links for Packages, Upload, Statistics, Documentation, Downloads, and Blog. A 'Sign in' link is located in the top right corner. The main heading reads 'Create .NET apps faster with NuGet'. Below this is a search bar with the placeholder text 'Search for packages...'. The central graphic features a stylized circuit board with three 3D cube-like components. Three data callouts are positioned around the graphic: '7,915,008 package versions' on the left, '547,898,813,208 package downloads' at the top, and '418,002 unique packages' on the right.

nuget Packages Upload Statistics Documentation Downloads Blog Sign in

Create .NET apps faster with NuGet

Search for packages...

7,915,008 package versions

547,898,813,208 package downloads

418,002 unique packages

NuGet

If you use Cider to manage your APL projects, you can use

```
]CIDER.AddNuGetDependencies
```

to add NuGet packages as dependencies in your project.

NuGet

```
dir←'/tmp/nuget-demo'
```

```
NuGet.Setup dir
```

```
Created project file "/tmp/nuget-demo/_nuget-demo.csproj"
```

```
NuGet.Add dir 'Clock/1.0.3'
```

```
Added/Updated: Clock --version 1.0.3
```

```
□USING←NuGet.Using dir
```

```
Clock.UtcNow
```

```
9/16/2024 11:02:04 PM +00:00
```

APLProcess

APLProcess is a cross-platform utility to start an APL interpreter in a new CPU process.

The `isolate` workspace uses APLProcess

```
p←NEW APLProcess 'myWs' 'maxws=4G'  
p←APLProcess.New 'myWs' 'maxWs=4G'  
p.RideInit←22361  A 'SERVE:*:22361'  
p.Run
```

APLProcess RPC Sample

```
A create and configure the instance of APLProcess
RPC_Process←APLProcess.New''
RPC_Process.Ws←□WSID                                A workspace to load
RPC_Process.Args←'RPC_Port=',⌞port                 A port to use for RPC
RPC_Process.Args,←'RPC_Threated=',⌞threaded       A run the server in a
                                                    thread in the child process?
RPC_Process.RunTime←runtime                          A use runtime interpreter?
RPC_Process.RideInit←''(port+1)rideable⇒~0 1⌞rideable A RIDE port
RPC_Process.Detach←~detach A don't kill child proc. when instance goes away

A try to launch the child process
:If 0≠⇒(rc msg)←RPC_Process.Run
```

APLProcess Uses

- ◆ Distribute processing to other processors (a la isolates)
- ◆ Offload processing
 - ◆ Spawn a lengthy process
 - ◆ Spawn a "risky" process
- ◆ Run on other machines via ssh (coming soon)

<https://dyalog.github.io/APLProcess>

APLProcess

Overview

User Guide

Reference

APLProcess Instance

Shared Methods

Samples

Sample Uses

About

License

Release Notes

Print

Overview

APLProcess is a cross-platform utility that enables you to easily start additional processes running Dyalog APL. This can be useful in a number of scenarios including:

- to distribute the processing load across multiple processors. This is what the isolate workspace does.
- to delegate processing to another process so that if there is some sort of failure, the primary process is not impacted.

APLProcess runs on Windows, Linux and MacOS. ssh support, which will enable you to start an APL process on another platform using the ssh protocol, is forthcoming.

Terminology

Parent Process

This is the process from where **APLProcess** is used to start "child" processes.

Child Process

This is a process started by a "parent" process. Each child process has a reference back to the "parent" process in the form of a "parent process ID" (PPID).

**On this page**

Terminology

Obtaining APLProcess

Dyalog APL Version 19.0 and later (using Link)

Dyalog APL versions before 19.0 (using SALT)

Upgrading to the Latest APLProcess

APLProcess is Available as a Tatin Package

OpenAI

- ◆ Started as a noodle
- ◆ It's a little like trying to change the oil in a moving car
- ◆ A subset of endpoints are implemented
 - ◆ More (including Assistants) are coming in October 2024
- ◆ Good HttpCommand usage model

<https://dyalog.github.io/OpenAI>

OpenAI

Overview

User Guide and Reference

Using OpenAI

Audio Endpoint

Chat Endpoint

Files Endpoint

Image Endpoint

Models Endpoint

Moderations Endpoint

About

License

Release Notes

Print

Overview

`OpenAI` is a Dyalog APL namespace that contains code to interact with the [OpenAI API](#). `OpenAI` develops and maintains several Large Language Models (LLMs). `OpenAI` contains code to interact with OpenAI endpoints. A few things to note:

- We are going to use the term "OpenAI" a lot in this documentation. When formatted as `OpenAI`, we are referring to the Dyalog namespace which implements an interface to the OpenAI API. When formatted as `OpenAI`, we are referring to the OpenAI API itself.
- ChatGPT is actively developing OpenAI, adding new LLMs and endpoint definitions. This puts us in a reactive mode to incorporate these new features and interfaces. We anticipate
- Not all OpenAI endpoints are currently implemented in `OpenAI`. This is partly due to the reason stated above. Our goal is to have sufficient endpoint coverage that our users can perform useful tasks with OpenAI from APL.
- There are other LLMs available - `OpenAI` should serve as a good model for how to implement interfaces to them. In particular, `OpenAI` makes heavy use of `HttpCommand` and the techniques used should be applicable for interacting with other LLM APIs.
- This documentation presents information on how to use the `OpenAI` namespace to interact with the OpenAI endpoints. It does not attempt to document all of the features and nuances of those endpoints. For that information, please see the [OpenAI API Reference](#).

On this page

Endpoints Currently Implemented

Forthcoming Endpoints

Expected in October 2024

Future

OpenAI Endpoints

- Models – return information about available LLMs
- Moderations – measure the "harmful" potential of text
- Audio – text to speech, transcription and translation
- Image – create and modify images based on prompts
- Chat – have a conversation with an OpenAI assistant

OpenAI.Models

- Returns information about available OpenAI models

```
m←OpenAI.Models
```

```
≠m.Data.data
```

```
31
```

```
6 2 p m.Data.data.id
```

```
tts-1-1106          dall-e-2  
gpt-4-1106-preview  tts-1  
tts-1-hd-1106      tts-1-hd  
dall-e-3            whisper-1  
text-embedding-3-large  text-embedding-3-small  
text-embedding-ada-002  gpt-4-turbo
```

OpenAI.Moderations.

- Moderation - Returns evaluation of "potentially harmful" text
- Demo

OpenAI.Audio.

Speech – text to audio

Transcription – audio to native text

Translation – audio to English text

OpenAI.Audio.Speech

Translate text to speech

OpenAI makes heavy use of `HttpCommand` and requires `HttpCommand` version 5.6 or later. During initialization, `OpenAI` will look for `HttpCommand` in its parent namespace and if it doesn't find `HttpCommand` it will load it from your Dyalog installation and then upgrade to the latest version of `HttpCommand`. The endpoints implemented in `OpenAI` will initialize `OpenAI` if not already initialized. You can also initialize `OpenAI` by running `OpenAI.Initialize`. In a production environment, `HttpCommand` should be copied and saved into the workspace rather than relying on loading and upgrading.

OpenAI.Audio.Speech

Translate text to speech

```
s←OpenAI.Audio.Speech myText  
s.OutFile←'/Dyalog24/NewPackages/speech'  
s.Run
```

```
[rc: 0 | msg: | HTTP Status: 200 "OK" | 740160 bytes  
written to c:/Dyalog24/NewPackages/speech.mp3]
```



OpenAI.Audio.Speech

Translate text to speech

```
s←OpenAI.Audio.Speech myText
```

OpenAI makes heavy use of `HttpCommand` and requires `HttpCommand` version 5.6 or later. During initialization, `OpenAI` will look for `HttpCommand` in its parent namespace and if it doesn't find `HttpCommand` it will load it from your Dyalog installation and then upgrade to the latest version of `HttpCommand`. The endpoints implemented in `OpenAI` will initialize `OpenAI` if not already initialized. You can also initialize `OpenAI` by running `OpenAI.Initialize`. In a production environment, `HttpCommand` should be copied and saved into the workspace rather than relying on loading and upgrading.

OpenAI.Audio.Transcription

- ◆ Transcribes audio speech to native text

"Welcome to Dyalog24. We hope you have fun and learn something new."

```
ts←OpenAI.Audio.Transcription inFile outFile  
ts.Run
```

OpenAI.Audio.Transcription

- ◆ Transcribes audio speech to native text

"Welcome to Dyalog24. We hope you have fun and learn something new."

Danish: "Velkommen til Dialog24. Vi håber I får det sjovt at lære noget nyt."

German: "Willkommen zu Dialog24. Wir hoffen, dass Sie Spaß haben und etwas Neues lernen werden."

Italian: "Benvenuti a Dialog24, vi auguriamo di divertirvi ed imparare qualcosa di nuovo!"

OpenAI.Audio.Translation

- ◆ Transcribes audio speech to English text

"Welcome to Dyalog24. We hope you have fun and learn something new."

```
tl←OpenAI.Audio.Translation inFile outFile
```

```
tl.Run
```

OpenAI.Audio.Translation

- ◆ Transcribes audio speech to English text

"Welcome to Dyalog24. We hope you have fun and learn something new."

Danish: "Welcome to Dialog24. We hope you have fun and learn something new."

German: "Welcome to Dialog24. We hope you will have fun and learn something new."

Italian: "Welcome to Dialog24! We wish you to have fun and learn something new!"

OpenAI . Image .

- Create – creates image(s) from prompt
- Edit – edit an existing image
- Vary – create variation(s) of an existing image
- Demo

OpenAI.Chat.

- Completion – have a chat with an assistant
- Demo

Into the future... Tatin-wise

- ◆ Dyalog will release additional Tatin packages as we identify good candidates
- ◆ Please share your priorities!
- ◆ Please share your own utilities as Tatin packages

Into the future... OpenAI-wise

- OpenAI is at version 0.1 and very much experimental
 - Please provide feedback
- We will try to keep up as OpenAI evolves

Thank you!

