

Parthorama

Parthorama.com is a general purpose app development tool which supports client-server apps. The apps are written in a new programming language called Parthonyte, as well as a text markup language called Parthotags. Parthonyte is implemented in Java. Client-server apps are hosted by third party Java hosting services. Apps run on Windows, Mac, and Linux.

Business Model

Subscribers pay \$10/year, and enjoy unrestricted use of client-server apps. Clients of nonprofit organizations always enjoy unrestricted use of client-server apps run by the organization they belong to, and like capped users, they pay no fees.

The amount of data capped users download and upload to a given client-server app in a given calendar month is capped. When the upload cap is reached, they can no longer upload any data (in excess of 80 bytes per user action). Similarly, when the download cap is reached, each download operation is limited to 5000 bytes of Parthotags code, and no data files can be downloaded.

Cap Amounts

The upload and download cap amounts are equal to the lowest quintile and highest quartile of bandwidth, respectively, for all users of a given client-server app in the previous calendar month. The bandwidth equals the amount of data downloaded and uploaded by a given user.

So when a capped user reaches the upload cap amount exceeded by 80 percent of the users of a given client-server app X in the previous calendar month, data uploads by that user to app X are restricted until the end of the month. Similarly, when a capped user reaches the download cap amount exceeded by 25 percent of the users of a given client-server app X in the previous calendar month, data downloads by that user from app X are restricted until the end of the month.

Quantity Discount

Organizations receive a quantity discount when paying subscription fees of multiple employees and customers. The fee charged to the organization is equal to \$20/year or \$5/user/year, whichever is higher.

AI Automates Coding

By the time Parthorama becomes operational, ample Parthonyte-based training data will exist. The AI LLMs will enable end-users to develop client-server apps, without knowledge of coding. So thanks to AI, the Parthorama community of app builders will flourish.

Steps

1. Develop foundation of Parthonyte code execution - **done!**
2. Pitch project idea to Progress Place
3. Develop rest of Parthonyte code execution: WCNMIL
 1. Wrap up core foundation features
 2. Classes and objects
 3. Non-scalar data types
 4. Modules
 5. Inheritance + Interfaces (hedrons)
 6. Library
4. Release Parthonyte as console-based compiler on GitHub
5. Begin recruiting contributors
6. Write Parthotags design specs
7. Develop Parthotags
8. Integrate Parthonyte with Parthotags
9. PYRE: Parthonyte Runtime Environment (open source)
10. Phase 1: monospaced text
11. Develop Parthonyte code editor
12. Expand code editor to Parthonyte SDK
13. Develop Parthonyte-to-Java converter
14. Create forums for developers and end-users
15. Develop desktop app store
16. Launch website
17. Purchase Google AdWords advertising
18. Lower priority features:
 1. Implement Keyboard Aid (bells and whistles of editor)
 2. Develop WYSIWYG Parthotags screen editor
19. Hire Waterloo co-op student or coder on autism spectrum
 1. Specialisterne finds IT jobs for those on autism spectrum
 2. Person hired is a Java Programmer
20. Java Programmer to handle Phase 2: variable-width text
21. Meanwhile, pitch project idea to DMZ tech incubator
22. If search for angel investor succeeds
23. Or, if grant proposal to Ministry of Health succeeds
24. Then Java Programmer becomes co-founder
25. Else Java Programmer is laid off
26. Monetize client-server apps
27. Design Progress Place intranet
28. Implement support for mobile devices:
 1. Hire Android/Swift programmer if sufficient funds
 2. Port PYRE to Android
 3. Develop mobile app store
 4. Convert PYRE for Android to Swift
 5. Develop PYRE for iOS from raw Swift codebase
 6. Develop optional code editor

Progress Place

Mike is a member of Progress Place, which is a clubhouse for consumer/survivors. Mike intends to pitch his Parthorama project idea to Progress Place. The idea is to create a Parthonyte-based intranet for Progress Place members and staff. Members can submit posts and photos, similar to a Facebook group. The intranet is an example of a Parthonyte-based client-server app for a nonprofit organization. After the Parthorama website is launched and the co-founder is hired, a search will commence for an angel investor. If that search is unsuccessful, a grant proposal will be submitted to the Ministry of Health. The intranet helps Progress Place engage with its members in a virtual, online community, complementing its existing bricks and mortar location. This saves money for the Ministry of Health, since Progress Place helps prevent costly hospitalizations due to its Clubhouse Model of psychosocial rehabilitation.

Monospaced Mode

Only monospaced mode is supported in Phase 1 (variable-width text is supported in Phase 2). All characters in a given panel are the same size, and adjacent cells in a given panel may be merged to form a subpanel. Panels and subpanels can contain a widget, graphic, or a block of text. Different panels containing text need not share the same font size. Subscripts and superscripts are offset vertically by half the height of a character cell. Lines of text can optionally be separated by a gap equal to half of a character cell. The canvas property of a panel supports drawing graphics such as lines, circles, and rectangles. Panels with canvases do not contain a character grid.

Edit Code on a Phone

Using nested menus, an onscreen "keyboard" has 3 to 5 rows of buttons/text. The top row includes icons for commonly used global functions, and the bottom row includes Space, Enter, Open/Close Parenthesis, and optionally Up/Down Arrow. The middle 1 to 3 rows display menu options: words or special characters, from left to right, then top to bottom. The Up/Down Arrow is used when the available menu options don't fit on one screen. A regular alphanumeric keyboard is displayed when entering identifier declarations. Scrolling vertically or horizontally is accomplished by swiping, and dynamic word wrap (default is enabled) ensures that the rightmost character of a line is always visible.

Default indentation: 2 spaces per level. Every program statement ends with a semicolon. Except for the first line, all lines of a multiline program statement are indented 1 space. Typing close parenthesis automatically inserts semicolon when needed. All text (except comments) is displayed as either upper or lower case, depending on the case-display switch, but that text is stored as lower case in the actual source file. By convention, identifiers of constant values terminated with underscore.

Java Converter

Converts Parthonyte code to server-side Java code. Parthonyte variable declarations are handled by using variable names beginning with an uppercase letter preceded by a type character: b for boolean, i for int, j for long, c for char, f for double, and s for string. Object variables followed by comment beginning with the class name.

Logo

Parentheses + Mountain + Pi = parenthesized version of the Python programming language, which is named after Monty Python = Mont(y) + Py(thon).

Glossary

Parthorama	Project name
Parthonyte	New programming language, similar to Python
Parthotags	Text markup language, similar to HTML
PYRE	Parthonyte Runtime Environment
.PTHY	Parthonyte source file
.PTHX	Parthonyte compiled unit
.POTG	Parthotags file
.POTJ	Parthotags/JSON file

About Us

I am Mike Hahn, the founder of Parthorama.com. I was previously employed at Brooklyn Computer Systems as a Delphi Programmer and a Technical Writer (I worked there between 1996 and 2013). At the end of 2014 I quit my job as a volunteer tutor at Fred Victor on Tuesday afternoons, where for 5 years I taught math, computers, and literacy, and became a volunteer math/computer tutor at West Neighbourhood House. I quit that job in mid-2019. I have a part-time job working for a perfume store. My hobbies are reading and I often go for walks. I don't read books very often, but on March 19, 2021 I started reading a biography of Steve Jobs which my brother gave me. I read the CBC news website, news/tech articles on my Flipboard app, and science/tech articles (under Google) on my phone. I visit my brother about once a month.

Contact Info

Mike Hahn
Phone: 416-533-4417
Email: hahnbytes (AT) gmail (DOT) com

Web: treenimation.net/hahnbytes/
Country: Canada

Parthonyte

Parthonyte (implemented in Java) is an open source Python dialect in which all operators precede their operands, and parentheses are used for all grouping (except string literals, which are delimited with double quotes, also statements are separated by semicolons). Parthonyte source files have a .PTHY extension. Parthotags is a text markup language, with a .POTG extension. Parthonyte boasts an ultra-simple Lisp-like syntax unlike all other languages.

Special Characters

Core:

- () grouping
- - word separator
- ; end of stmt.
- : dot operator
- " string delimiter
- \ escape char.

Operators:

- + - * / %
- = < >
- & | ^ ~ ! ?

Other:

- # comment
- {} block comment
- _ used in identifiers
- \$ string prefix char.

Differences from Python

- Parentheses, not whitespace
- Operators come before their operands
- Integration with Parthotags
- Information hiding: public/private
- Single, not multiple inheritance
- Adds interfaces: "hedron" defs.
- Drops iterators and generators
- Adds lambdas
- Adds quote and list-compile functions, treating code as data
- Adds cons, car and cdr functionality

Parthotags

Parthotags is a simplified markup language used to replace HTML. Mock JSON files using Parthotags syntax have a .POTJ extension, and include no commas. Instead of myid: val, use [myid: val]. Instead of [1, 2, 3], use [arr: [: 1][: 2][: 3]]. Arbitrary Parthotags code can be embedded in the Parthonyte echo statement. Parthotags syntax, where asterisk (*) means occurs zero or more times, is defined as follows:

Tags:

- [tag]
- [tag (fld val)*: body]
- [tag (fld val)*| body | tag]

Body:

- text
- [(fld val)*: text]*

Parthonyte call:

- [expr: <expr>]
- [exec: <stmt>...]
- [pthy: <path>]

Note: for fld = style, corresponding val = (fld val)*

Benefits of Parthonyte

Parthonyte is simpler than any other object-oriented programming language, and integrates nicely with Parthotags, which is simpler than HTML. Operators come before their operands, also statements are semicolon-separated, so they never start with a parenthesis. Parthonyte is written in Java, so apps can be written in one or both of the Java and Parthonyte programming languages. Parthotags is simpler than HTML, based on nested rows and columns. A row cell is divided into multiple variable-width column cells, and a column cell is divided into multiple variable-height row cells.

Keyboard Aid

The close delim switch of the Parthonyte code editor enables the automatic insertion of a closing parenthesis, brace, or double quote whenever the open delimiter is inserted. The optional keyboard-aid feature enables hyphens, open parentheses, and close parentheses to be entered by typing semicolons, commas, and periods, respectively. When enabled, keyboard-aid can be temporarily suppressed by using the Ctrl key in conjunction with typing semicolons, commas, and periods (no character substitution takes place).

By convention, hyphens are used to separate words in multi-word identifiers, but semicolons are easier to type than hyphens. Similarly, commas and periods are easier to type than parentheses. Typing semicolon converts previous hyphen to a semicolon, and previous semicolon to a hyphen (use the Ctrl key to override this behaviour). Typing semicolon after close parenthesis simply inserts semicolon. Typing space after hyphen at end of identifier converts hyphen to underscore.

More...

Please click on More to access miscellaneous Parthonyte documentation. Only the first 2 paragraphs of the More web page contain up-to-date info, the rest of that web page is obsolete.