|  |
| --- |
| Circle Language Spec |

## Ideas (Out of Scope)

**[ Preliminary documentation ]**

Software System, Projects,

(Old idea. At 2008-09-01 I stated that it was an old idea.)

Ik denk dat ik zelfs bij de projectenlijst eerst voor het compleet maken van de bruikbaarheid van de taal ga. Dat wil zeggen: permanent storage erbij.

Daarna voor de integratie van vooral concepts, en mogelijk literals.

En eventueel andere fundamentele invalshoeken erbij.

En mogelijk voor diagrammen erin te integreren op een of andere manier, eventueel, als ik het qua volledige overlap in codeer concepten en definitie van de diagramtaal niet al voldoende 1 taal vind.

JJ

-----

Software System,

Na het documenteren wil ik wel snel resultaten zien.

De ontwerpprojecten zullen wat langer duren.

Maar bij de ontwikkelprojecten wil ik heel snel resultaten zien. Er zijn bepaalde fundamentele dingen, die veranderen aan m'n systeem. Maar ik wil niet eerst 100 fundamentele dingen ontwikkelen, voordat ik resultaat zie.

Wie weet is het ontwerp straks zo kraakhelder, dat het niet moeilijk meer is om de enigszins fundamentele dingen te implementeren. Wie weet sla ik gewoon wat fundamentele dingen over, omdat ik minder fundamentele dingen toch eerder wil zien.

Er zullen verschillende soorten ontwikkelprojecten zijn:

- Fundamenteel

- Programmeertaal

- Facilitair / applicaties

Ik wil geen 100 fundamentele dingen eerst ontwikkelen.

JJ

-----

Software System,

De OS kernel is zeg maar niet mijn core business.

Niet dat je echt van een core business kan spreken.

Maar daarom heb ik er niet zo'n zin in om er veel tijd in te steken. Grappig, dat ik die modules tezamen wel de Core noem...

JJ

-----

Computer Language,

2008-09-15

You might want projects about Internet as a Single Computer, about cached storage and the connection between the logical and the physical.

AND transparent security.

AND concurrency resolution.

They may be needed for this language to succeed. But they are not worked out yet at all.

JJ

-----

Computer Language,

2008-09-15

A possible aim in a broader perspective might be that low-level device driver programming, and high-level programming for a world-wide network of computer is done through a single same paradigm, without loss of performance.

JJ

-----

Computer Language,

However, Compared Input Output and Automatic Execution Order is only ONE candidate for concurrency resolution. It is totally not clear yet how the concurrency resolution mechanism will look. This makes me nervous. I am several subjects away from having worked out all concurrency resolution candidates, and it is vital for Internet as a Single Computer. And Internet as a Single Computer is vital for the language to succeed.

JJ

-----

Document Software System,

2009-04-27

Perhaps one day it is also important to mention how original the ideas are supposed to be.

JJ

-----

New Computer Language,

2010-01-27

Another reason the name Circle might be nice, because if you'd use the language as a user interface, it is a non-square user interface. Most user interfaces are square oriented or rectangularly organized. But Circle is not. It is non-square.

JJ

-----

New Computer Language Spec,

2009-04-20

You already have those things, even when you do not have Concurrency Resolution or Internet as a Single Computer or have incorporated existing ideas of other people. That's why it is so important to first do the easy projects.

JJ

-----

Circle Language Spec,

In articles’ See also references, you can have backward references and foreward references. Maybe references to other articles are not that important, though.

Possibly move globality higher up, before Commands, and mention in Commands, that they are usually made part of a globality.

JJ

-----

Circle Language Spec,

2008-09-01

Near the end of the development of the new programming language, next to the application programming projects, you might have a set of extreme test cases, complete, workable and fast enough. These are extreme test cases, that you might want possible in the new general purpose programming language.

It are the ones for which exists doubt, that the first version will be able to handle, so consider a challenge, especially in the area of performance and complete for the purpose it will be used for:

- Real-time music synthesizer

- Game-speed graphics rendering

- Device driver pack

- Network-distributed application

- Business process automation

- Navigate existing systems over the internet

- Playing video

(Does not look like test cases, more like complete applications. Maybe just a prototype could be a test case.)

Perhaps even further down the road, you should have even more extreme test cases:

- Real-time distributed music synthesizer

- Game-speed distributed graphics rendering

- Auto-hack weak spots in computer networks

JJ

-----

Circle Language Spec,

2008-09-15

I am currently working on the concept of parameters, and it has small scope compared to something like Internet as a Single Computer. I am beginning to get nervous about it, that I am focusing on a detail so much, and not working on the broader view of things.

However, parameters are kind of important. Because they might be the base for concurrency resolution with the help of Command Input Output and Automatic Execution Order.

JJ

-----

Circle Language Spec,

2009-05-13

The Fundamental Principles makes comparisons to other systems. However, not comparing is a writing style rule chosen later. However, the Fundamental Principles may be an exception: it explains why this language is there. And the new language might solve a problem in another language, hence comparison may be required.

JJ