## Subject: "A picture paints a thousand words"
Posted by Anonymous on Mon, 27 Apr 2020 12:18:39 GMT

View Forum Message <> Reply to Message

Originally posted by: Gareth Evans

For some years I'd been toying with the idea of
a pictorial method of entering and maintaining
programs, bearing in mind the thread title.

With millions of frustrated programmers around the
World I am sure that I am not the only person to
be thinking in that direction, "There's nothing
new under the Sun".

I'd intended to base my stuff around the structured
flowcharts in the (now discontinued) British standard
of BS6224.

With a relative of mine being a schoolteacher, I've just
had my eyes opened to the Scratch language.

Wow!

## Subject: Re: "A picture paints a thousand words"
Posted by Ahem A Rivet's Shot on Mon, 27 Apr 2020 13:18:19 GMT

View Forum Message <> Reply to Message

On Mon, 27 Apr 2020 13:18:39 +0100
Gareth Evans <headstone255@yahoo.com> wrote:

> For some years I'd been toying with the idea of
> a pictorial method of entering and maintaining
> programs, bearing in mind the thread title.

 It's been tried a few times - the usual result is dialogue box
laden, frustrating and limited in scope.

> I'd intended to base my stuff around the structured
> flowcharts in the (now discontinued) British standard
> of BS6224.

 Nobody uses flowcharts these days.

--
Steve O'Hara-Smith                      |   Directable Mirror Arrays
C:\>WIN                                 | A better way to focus the sun

The computer obeys and wins.        |    licences available see
You lose and Bill collects.         |    http://www.sohara.org/

---

## Subject: Re: "A picture paints a thousand words"
Posted by Anonymous on Mon, 27 Apr 2020 14:56:09 GMT
View Forum Message <> Reply to Message

Originally posted by: Johann 'Myrkraverk' Oskarsson

On 27/04/2020 8:18 pm, Gareth Evans wrote:
> For some years I'd been toying with the idea of
> a pictorial method of entering and maintaining
> programs, bearing in mind the thread title.

I seem to recall something like a picture programming language from
Novell in the 90s.  I either had a limited (free?) edition that didn't
create executables, or it didn't do what I thought it did.

I never came across it again, nor do I remember what it was called.  It
could have been from someone other than Novell.  Apparently, a quick web
search doesn't turn anything up.

So, it's been tried before, and hasn't caught on.


--
Johann | email: invalid -> com | www.myrkraverk.com/blog/
I'm not from the Internet, I just work there. | twitter: @myrkraverk

---

## Subject: Re: "A picture paints a thousand words"
Posted by Anonymous on Mon, 27 Apr 2020 16:04:53 GMT
View Forum Message <> Reply to Message

Originally posted by: Gareth Evans

On 27/04/2020 14:18, Ahem A Rivet's Shot wrote:
> On Mon, 27 Apr 2020 13:18:39 +0100
> Gareth Evans <headstone255@yahoo.com> wrote:

>
>> I'd intended to base my stuff around the structured
>> flowcharts in the (now discontinued) British standard
>> of BS6224.
>
> Nobody uses flowcharts these days.

>

Then I'm nobody. An algorithm designed gaphically is
expressable in any language.

---

## Subject: Re: "A picture paints a thousand words"
Posted by Anonymous on Mon, 27 Apr 2020 16:08:44 GMT
View Forum Message <> Reply to Message

Originally posted by: Gareth Evans

On 27/04/2020 15:56, Johann 'Myrkraverk' Oskarsson wrote:
> On 27/04/2020 8:18 pm, Gareth Evans wrote:
>> For some years I'd been toying with the idea of
>> a pictorial method of entering and maintaining
>> programs, bearing in mind the thread title.
>
> I seem to recall something like a picture programming language from
> Novell in the 90s.  I either had a limited (free?) edition that didn't
> create executables, or it didn't do what I thought it did.
>
> I never came across it again, nor do I remember what it was called.  It
> could have been from someone other than Novell.  Apparently, a quick web
> search doesn't turn anything up.
>
> So, it's been tried before, and hasn't caught on.
>
>

Well, what I was thinking of was something that would produce
executable code but without first vomiting up some C, FORTRAN,
ALGOL, assembler, or what ever floats your boat.

This afternoon I have been browsing Scratch and Blocky and
discovered by googling other VPLs. That' the trouble of being
retired without day-to-day contact with other softies; you
don't get knowledge of developments.

---

## Subject: Re: "A picture paints a thousand words"
Posted by scott on Mon, 27 Apr 2020 16:13:24 GMT
View Forum Message <> Reply to Message

Ahem A Rivet's Shot <steveo@eircom.net> writes:
> On Mon, 27 Apr 2020 13:18:39 +0100
> Gareth Evans <headstone255@yahoo.com> wrote:

>
>> For some years I'd been toying with the idea of
>> a pictorial method of entering and maintaining
>> programs, bearing in mind the thread title.
>
>  It's been tried a few times - the usual result is dialogue box
> laden, frustrating and limited in scope.
>
>> I'd intended to base my stuff around the structured
>> flowcharts in the (now discontinued) British standard
>> of BS6224.
>
> Nobody uses flowcharts these days.

Please don't feed the troll.

---

## Subject: Re: "A picture paints a thousand words"
Posted by Anonymous on Mon, 27 Apr 2020 16:19:11 GMT
View Forum Message <> Reply to Message

Originally posted by: Gareth Evans

On 27/04/2020 17:13, Scott Lurndal wrote:
>
> Please don't feed the troll.
>

We have crossed swords before because of your
immature outbursts such as is quoted above.

Grow up, Sonny.

---

## Subject: Re: "A picture paints a thousand words"
Posted by David LaRue on Mon, 27 Apr 2020 16:39:39 GMT
View Forum Message <> Reply to Message

Gareth Evans <headstone255@yahoo.com> wrote in
news:r8702t$6m7$1@dont-email.me:

> On 27/04/2020 15:56, Johann 'Myrkraverk' Oskarsson wrote:
>> On 27/04/2020 8:18 pm, Gareth Evans wrote:
>>> For some years I'd been toying with the idea of
>>> a pictorial method of entering and maintaining
>>> programs, bearing in mind the thread title.
>>

>> I seem to recall something like a picture programming language from
>> Novell in the 90s.  I either had a limited (free?) edition that
>> didn't create executables, or it didn't do what I thought it did.
>>
>> I never came across it again, nor do I remember what it was called.
>> It could have been from someone other than Novell.  Apparently, a
>> quick web search doesn't turn anything up.
>>
>> So, it's been tried before, and hasn't caught on.
>>
>>
>
> Well, what I was thinking of was something that would produce
> executable code but without first vomiting up some C, FORTRAN,
> ALGOL, assembler, or what ever floats your boat.
>
> This afternoon I have been browsing Scratch and Blocky and
> discovered by googling other VPLs. That' the trouble of being
> retired without day-to-day contact with other softies; you
> don't get knowledge of developments.

I looked at Scratch several years ago.  I program in many languages but
C/C++ the most.  I had to restructure some code this morning to fit a
new data model.  While it could be flowcharted or displayed visually the
challenge is also to allow the developer to work with a creation and
create something new.  Scratch allows some restructuring but needs more
to help the developer.  Building big or real world applications or
systems in Scratch would likely take too long and be unmaintainable.

It is worthy to consider such problems though.  It makes us think and
strive to create better ways.  There was a long thread in one of the
system modeling groups many years ago by a retired professional that was
teaching how to create everything from the requirements.   He was a
wonderful teacher.

David

---

Subject: Re: "A picture paints a thousand words"
Posted by Peter Flass on Mon, 27 Apr 2020 17:51:56 GMT
View Forum Message <> Reply to Message

Ahem A Rivet's Shot <steveo@eircom.net> wrote:
> On Mon, 27 Apr 2020 13:18:39 +0100
> Gareth Evans <headstone255@yahoo.com> wrote:
>
>> For some years I'd been toying with the idea of
>> a pictorial method of entering and maintaining

>> programs, bearing in mind the thread title.
>
> It's been tried a few times - the usual result is dialogue box
> laden, frustrating and limited in scope.
>
>> I'd intended to base my stuff around the structured
>> flowcharts in the (now discontinued) British standard
>> of BS6224.
>
> Nobody uses flowcharts these days.
>

I still do, very occasionally, when the the task lends itself to them. I
also have used hierarchy charts, decision tables, and other "obsolete"
techniques. It's good to know how to use them.

--
Pete

---

## Subject: Re: "A picture paints a thousand words"
Posted by Anonymous on Mon, 27 Apr 2020 19:23:55 GMT

Originally posted by: Bob Eager

On Mon, 27 Apr 2020 10:51:56 -0700, Peter Flass wrote:

>> Nobody uses flowcharts these days.
>>
> I still do, very occasionally, when the the task lends itself to them. I
> also have used hierarchy charts, decision tables, and other "obsolete"
> techniques. It's good to know how to use them.

I mainly use pencil and paper for state diagrams!


--
Using UNIX since v6 (1975)...

Use the BIG mirror service in the UK:
 http://www.mirrorservice.org

---

## Subject: Re: "A picture paints a thousand words"
Posted by Anonymous on Mon, 27 Apr 2020 20:38:27 GMT

Originally posted by: J. Clarke

On Mon, 27 Apr 2020 10:51:56 -0700, Peter Flass
<peter_flass@yahoo.com> wrote:

> Ahem A Rivet's Shot <steveo@eircom.net> wrote:
>> On Mon, 27 Apr 2020 13:18:39 +0100
>> Gareth Evans <headstone255@yahoo.com> wrote:
>>
>>> For some years I'd been toying with the idea of
>>> a pictorial method of entering and maintaining
>>> programs, bearing in mind the thread title.
>>
>>   It's been tried a few times - the usual result is dialogue box
>> laden, frustrating and limited in scope.
>>
>>>  I'd intended to base my stuff around the structured
>>>  flowcharts in the (now discontinued) British standard
>>>  of BS6224.
>>
>>   Nobody uses flowcharts these days.
>>
>
> I still do, very occasionally, when the the task lends itself to them. I
> also have used hierarchy charts, decision tables, and other "obsolete"
> techniques. It's good to know how to use them.

When I'm trying to understand 40 year old spaghetti APL, I find that
flowcharting it is very beneficial.  If nothing else, a look at the
flowchart gives my manager an idea of what I'm dealing with.

---

## Subject: Re: "A picture paints a thousand words"
Posted by Anonymous on Mon, 27 Apr 2020 20:39:20 GMT

Originally posted by: drb

>        Nobody uses flowcharts these days.

Or, apparently, any other form of design documentation.  That's
not a good thing.

De

Subject: Re: "A picture paints a thousand words"
Posted by Anonymous on Mon, 27 Apr 2020 20:43:41 GMT

Originally posted by: J. Clarke

On Mon, 27 Apr 2020 22:56:09 +0800, Johann 'Myrkraverk' Oskarsson
<johann@myrkraverk.invalid> wrote:

> On 27/04/2020 8:18 pm, Gareth Evans wrote:
>>  For some years I'd been toying with the idea of
>>  a pictorial method of entering and maintaining
>>  programs, bearing in mind the thread title.
>
> I seem to recall something like a picture programming language from
> Novell in the 90s.  I either had a limited (free?) edition that didn't
> create executables, or it didn't do what I thought it did.
>
> I never came across it again, nor do I remember what it was called.  It
> could have been from someone other than Novell.  Apparently, a quick web
> search doesn't turn anything up.
>
> So, it's been tried before, and hasn't caught on.

Google "DRAKON programming language".

Another one, that doesn't use a flowchart paradigm but does involve
programming by dragging blocks around is "Scratch".

Subject: Re: "A picture paints a thousand words"
Posted by Anonymous on Mon, 27 Apr 2020 20:48:57 GMT

Originally posted by: J. Clarke

On Mon, 27 Apr 2020 16:43:41 -0400, J. Clarke
<jclarke.873638@gmail.com> wrote:

> On Mon, 27 Apr 2020 22:56:09 +0800, Johann 'Myrkraverk' Oskarsson
> <johann@myrkraverk.invalid> wrote:
>
>> On 27/04/2020 8:18 pm, Gareth Evans wrote:
>>>  For some years I'd been toying with the idea of
>>>  a pictorial method of entering and maintaining
>>>  programs, bearing in mind the thread title.
>>
>> I seem to recall something like a picture programming language from

>> Novell in the 90s.  I either had a limited (free?) edition that didn't
>> create executables, or it didn't do what I thought it did.
>>
>> I never came across it again, nor do I remember what it was called.  It
>> could have been from someone other than Novell.  Apparently, a quick web
>> search doesn't turn anything up.
>>
>> So, it's been tried before, and hasn't caught on.
>
> Google "DRAKON programming language".
>
> Another one, that doesn't use a flowchart paradigm but does involve
> programming by dragging blocks around is "Scratch".

Also, see today's (April 27,  2020) Google Doodle.

---

## Subject: Re: "A picture paints a thousand words"
Posted by Anonymous on Mon, 27 Apr 2020 20:49:34 GMT
View Forum Message <> Reply to Message

Originally posted by: Bob Eager

On Mon, 27 Apr 2020 16:43:41 -0400, J. Clarke wrote:

>  On Mon, 27 Apr 2020 22:56:09 +0800, Johann 'Myrkraverk' Oskarsson
> <johann@myrkraverk.invalid> wrote:
>
>> On 27/04/2020 8:18 pm, Gareth Evans wrote:
>>>  For some years I'd been toying with the idea of a pictorial method of
>>>  entering and maintaining programs, bearing in mind the thread title.
>>
>> I seem to recall something like a picture programming language from
>> Novell in the 90s.  I either had a limited (free?) edition that didn't
>> create executables, or it didn't do what I thought it did.
>>
>> I never came across it again, nor do I remember what it was called.  It
>> could have been from someone other than Novell.  Apparently, a quick web
>> search doesn't turn anything up.
>>
>> So, it's been tried before, and hasn't caught on.
>
>  Google "DRAKON programming language".
>
>  Another one, that doesn't use a flowchart paradigm but does involve
>  programming by dragging blocks around is "Scratch".

Scratch is used a lot in primary schools. I've been training primary

school teachers in computational thinking (for them to pass on to
pupils), and a lot of them use it.

--
Using UNIX since v6 (1975)...

Use the BIG mirror service in the UK:
 http://www.mirrorservice.org

---

Subject: Re: "A picture paints a thousand words"
Posted by Ahem A Rivet's Shot on Mon, 27 Apr 2020 21:22:54 GMT
View Forum Message <> Reply to Message

On Mon, 27 Apr 2020 15:39:20 -0500
drb@ihatespam.msu.edu (Dennis Boone) wrote:

>>       Nobody uses flowcharts these days.
>
> Or, apparently, any other form of design documentation.  That's
> not a good thing.

 Design documentation has been required to be written and reviewed
everywhere I've worked, it just doesn't include flowcharts largely because
they're insufficiently expressive. Anything simple enough to flowchart is
usually off the shelf in a well tested library.

--
Steve O'Hara-Smith            |   Directable Mirror Arrays
C:\>WIN                       | A better way to focus the sun
The computer obeys and wins.  |    licences available see
You lose and Bill collects.   |    http://www.sohara.org/

---

Subject: Re: "A picture paints a thousand words"
Posted by Peter Flass on Mon, 27 Apr 2020 21:58:01 GMT
View Forum Message <> Reply to Message

Dennis Boone <drb@ihatespam.msu.edu> wrote:
>>       Nobody uses flowcharts these days.
>
> Or, apparently, any other form of design documentation.  That's
> not a good thing.
>
> De
>

+1

--
Pete

---

Subject: Re: "A picture paints a thousand words"
Posted by Dan Espen on Mon, 27 Apr 2020 22:32:39 GMT
View Forum Message <> Reply to Message

drb@ihatespam.msu.edu (Dennis Boone) writes:

>>        Nobody uses flowcharts these days.
>
> Or, apparently, any other form of design documentation.  That's
> not a good thing.

Hmm, up until I retired, I wrote tons of documentation.
Back when this was on paper,
one person I know referred to them as "rare books".
We did a design that filled 8-10 loose leafs.

My last job I founds dozens of text and script files.
There was a main document and dozens of enhancement
descriptions.  I rewrote everything I could find as HTML.
I ended up with maybe 5000 lines of prose.
I used tables whenever I had a list,
that might have been 5% of the written material.

Out of all that, I only had 2 hierarchy charts which
I changed from ASCII art to SVG.  5 or 6 boxes in each.
I only left them because they were there when I started.

I'm not a big chart fan, you can say "A" calls "B" or
you can put "A" and "B" in boxes and connect them with an arrow.
Pretty much that same thing.

Flowcharts were the thing in the 60s and 70s.
When it turned out you could run your COBOL through a
program and produce a flow chart, flow charts fell out
of favor.  Pseudo code was quite popular in the 70s or
maybe 80s but I was never a fan.  Now I look at all of that
as a way of coding, so I just write the code liberally
sprinkled with comments about what I haven't written yet.

--
Dan Espen

## Subject: Re: "A picture paints a thousand words"
Posted by Anonymous on Tue, 28 Apr 2020 00:36:03 GMT

Originally posted by: J. Clarke

On 27 Apr 2020 20:49:34 GMT, Bob Eager <news0073@eager.cx> wrote:

> On Mon, 27 Apr 2020 16:43:41 -0400, J. Clarke wrote:
>
>> On Mon, 27 Apr 2020 22:56:09 +0800, Johann 'Myrkraverk' Oskarsson
>> <johann@myrkraverk.invalid> wrote:
>>
>>> On 27/04/2020 8:18 pm, Gareth Evans wrote:
>>>> For some years I'd been toying with the idea of a pictorial method of
>>>> entering and maintaining programs, bearing in mind the thread title.
>>>
>>> I seem to recall something like a picture programming language from
>> Novell in the 90s.  I either had a limited (free?) edition that didn't
>> create executables, or it didn't do what I thought it did.
>>>
>>> I never came across it again, nor do I remember what it was called.  It
>> could have been from someone other than Novell.  Apparently, a quick web
>> search doesn't turn anything up.
>>>
>>> So, it's been tried before, and hasn't caught on.
>>
>> Google "DRAKON programming language".
>>
>> Another one, that doesn't use a flowchart paradigm but does involve
>> programming by dragging blocks around is "Scratch".
>
> Scratch is used a lot in primary schools. I've been training primary
> school teachers in computational thinking (for them to pass on to
> pupils), and a lot of them use it.

Harvard has a computer science course for non-CS majors, "CS 50". They
start out with Scratch.

---

## Subject: Re: "A picture paints a thousand words"
Posted by Anonymous on Tue, 28 Apr 2020 02:10:35 GMT

Originally posted by: Radey Shouman

Gareth Evans <headstone255@yahoo.com> writes:

> On 27/04/2020 15:56, Johann 'Myrkraverk' Oskarsson wrote:
>> On 27/04/2020 8:18 pm, Gareth Evans wrote:
>>> For some years I'd been toying with the idea of
>>> a pictorial method of entering and maintaining
>>> programs, bearing in mind the thread title.
>>
>> I seem to recall something like a picture programming language from
>> Novell in the 90s. I either had a limited (free?) edition that didn't
>> create executables, or it didn't do what I thought it did.
>>
>> I never came across it again, nor do I remember what it was called. It
>> could have been from someone other than Novell. Apparently, a quick web
>> search doesn't turn anything up.
>>
>> So, it's been tried before, and hasn't caught on.
>>
>>
>
> Well, what I was thinking of was something that would produce
> executable code but without first vomiting up some C, FORTRAN,
> ALGOL, assembler, or what ever floats your boat.
>
> This afternoon I have been browsing Scratch and Blocky and
> discovered by googling other VPLs. That' the trouble of being
> retired without day-to-day contact with other softies; you
> don't get knowledge of developments.

Matlab simulink seems to be fairly popular these days. It's essentially
a pictorial (schematic) simulation language. Depends heavily on buying
expensive simulation blocks from the Mathworks, naturally.

---

Subject: Re: "A picture paints a thousand words"
Posted by Anonymous on Tue, 28 Apr 2020 02:24:05 GMT
View Forum Message <> Reply to Message

Originally posted by: Radey Shouman

David LaRue <huey.dll@tampabay.rr.com> writes:

> Gareth Evans <headstone255@yahoo.com> wrote in
> news:r8702t$6m7$1@dont-email.me:
>
>> On 27/04/2020 15:56, Johann 'Myrkraverk' Oskarsson wrote:
>>> On 27/04/2020 8:18 pm, Gareth Evans wrote:
>>>> For some years I'd been toying with the idea of
>>>> a pictorial method of entering and maintaining
>>>> programs, bearing in mind the thread title.

>>>
>>> I seem to recall something like a picture programming language from
>>> Novell in the 90s. Â  I either had a limited (free?) edition that
>>> didn't create executables, or it didn't do what I thought it did.
>>>
>>> I never came across it again, nor do I remember what it was called. Â
>>> It could have been from someone other than Novell. Â  Apparently, a
>>> quick web search doesn't turn anything up.
>>>
>>> So, it's been tried before, and hasn't caught on.
>>>
>>>
>>
>> Well, what I was thinking of was something that would produce
>> executable code but without first vomiting up some C, FORTRAN,
>> ALGOL, assembler, or what ever floats your boat.
>>
>> This afternoon I have been browsing Scratch and Blocky and
>> discovered by googling other VPLs. That' the trouble of being
>> retired without day-to-day contact with other softies; you
>> don't get knowledge of developments.
>
> I looked at Scratch several years ago.  I program in many languages but
> C/C++ the most.  I had to restructure some code this morning to fit a
> new data model.  While it could be flowcharted or displayed visually the
> challenge is also to allow the developer to work with a creation and
> create something new.  Scratch allows some restructuring but needs more
> to help the developer.  Building big or real world applications or
> systems in Scratch would likely take too long and be unmaintainable.
>
> It is worthy to consider such problems though.  It makes us think and
> strive to create better ways.  There was a long thread in one of the
> system modeling groups many years ago by a retired professional that was
> teaching how to create everything from the requirements.   He was a
> wonderful teacher.

Ladder logic is still quite popular for programming PLCs, it is even
standardized as IEC 61131-3.  Can't get much more graphical than that.

---

Subject: Re: "A picture paints a thousand words"
Posted by Ahem A Rivet's Shot on Tue, 28 Apr 2020 05:40:47 GMT
View Forum Message <> Reply to Message

On Mon, 27 Apr 2020 10:51:56 -0700
Peter Flass <peter_flass@yahoo.com> wrote:

> Ahem A Rivet's Shot <steveo@eircom.net> wrote:

>>    Nobody uses flowcharts these days.
>>
>
> I still do, very occasionally, when the the task lends itself to them. I

 It has been a long time since I've had a task that lent itself to a
flowchart.

> also have used hierarchy charts, decision tables, and other "obsolete"
> techniques. It's good to know how to use them.

 Yes it is, although any design I'm involved in is more likely to
require documenting language choice(s) (and reason(s) for it), a set of
class hierarchies, some service APIs, key algorithms, a test plan (what's
covered by unit testing and automated functional testing and what can only
be tested manually), a review of alternative solutions and an effort
estimate.

--
Steve O'Hara-Smith                |  Directable Mirror Arrays
C:\>WIN                           | A better way to focus the sun
The computer obeys and wins.           |   licences available see
You lose and Bill collects.       |   http://www.sohara.org/

---

Subject: Re: "A picture paints a thousand words"
Posted by Anonymous on Tue, 28 Apr 2020 08:51:21 GMT
View Forum Message <> Reply to Message

Originally posted by: Bob Eager

On Mon, 27 Apr 2020 20:36:03 -0400, J. Clarke wrote:

> On 27 Apr 2020 20:49:34 GMT, Bob Eager <news0073@eager.cx> wrote:
>
>> On Mon, 27 Apr 2020 16:43:41 -0400, J. Clarke wrote:
>>
>>> On Mon, 27 Apr 2020 22:56:09 +0800, Johann 'Myrkraverk' Oskarsson
>>> <johann@myrkraverk.invalid> wrote:
>>>
>>>> On 27/04/2020 8:18 pm, Gareth Evans wrote:
>>>> > For some years I'd been toying with the idea of a pictorial method
>>>> > of entering and maintaining programs, bearing in mind the thread
>>>> > title.
>>>>
>>>> I seem to recall something like a picture programming language from
>>>> Novell in the 90s.  I either had a limited (free?) edition that didn't

>>>> create executables, or it didn't do what I thought it did.
>>>>
>>>> I never came across it again, nor do I remember what it was called.
>>>> It could have been from someone other than Novell.  Apparently, a
>>>> quick web search doesn't turn anything up.
>>>>
>>>> So, it's been tried before, and hasn't caught on.
>>>
>>>  Google "DRAKON programming language".
>>>
>>>  Another one, that doesn't use a flowchart paradigm but does involve
>>>  programming by dragging blocks around is "Scratch".
>>
>> Scratch is used a lot in primary schools. I've been training primary
>> school teachers in computational thinking (for them to pass on to
>> pupils), and a lot of them use it.
>
>  Harvard has a computer science course for non-CS majors, "CS 50". They
>  start out with Scratch.

Interesting. And in the UK we teach it to 6 year olds!

--
Using UNIX since v6 (1975)...

Use the BIG mirror service in the UK:
 http://www.mirrorservice.org

---

Subject: Re: "A picture paints a thousand words"
Posted by Ahem A Rivet's Shot on Tue, 28 Apr 2020 09:04:34 GMT
View Forum Message <> Reply to Message

On 28 Apr 2020 08:51:21 GMT
Bob Eager <news0073@eager.cx> wrote:

>  Interesting. And in the UK we teach it to 6 year olds!

 OTOH the BBC had to censor material (9pm watershed) the Dutch teach
to 6 year olds in school!

--
Steve O'Hara-Smith                |   Directable Mirror Arrays
C:\>WIN                          | A better way to focus the sun
The computer obeys and wins.       |   licences available see
You lose and Bill collects.        |   http://www.sohara.org/