

Re: C# Nullable types

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework/2007-04/msg00119.html>

- *From:* "Patrice" <<http://www.chez.com/scribe/>>
 - *Date:* Thu, 5 Apr 2007 18:22:18 +0200
-

I will just end with how the SQL Server doc explains this.

<http://msdn2.microsoft.com/en-us/library/ms191504.aspx> :

"Null values generally indicate data that is unknown, not applicable, or that the data will be added later."

I'm not sure what is the point you don't like in this definition that looks quite close to mine but it really looks like that understanding each other on this topic is out of reach.

Take care and see you soon in another thread for another heated discussion ;-)

--

Patrice

"Scott M." <s-mar@xxxxxxxxxxxxxx> a écrit dans le message de news: OMPOZW5dHHA.4308@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

I think you are making my point here. You introduced the concepts and terms: "unknown", "undefined" and "not applicable" into the disussion. In addition, you are talking about "why" a database needs the concept of a null (I personally think you mis-stated that, but that is besides the point).

Below, you are talking about the practical use of null, rather than the meaning of null, which is the point of this thread.

"Patrice" <<http://www.chez.com/scribe/>> wrote in message news:ecXux94dHHA.4032@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

I'm not using null when "I don't know what value the user may want to input" (actually I never know what value the user will enter, not sure what you meant).

Also I don't use NULL when "I'm letting the user not put any value at all into the field" which is IMO bad form at least to do systematically. I'll

Re: C# Nullable types

use an empty string or 0 as a default value if it makes senses.

I will use this for example for the actual start date of something that not started yet. In this case you can't provide any value. IMO this is a more convincing sample than Appartement number and the like where in a real world scenario you never want to distinguish between a value that can't be provided for some reason (null) and the fact that the apartment number is just an empty string because you *know* this is an empty string, not because you are unable to provide this value.

This is perhaps why I tried at first to explain what's behind null. I felt that using simply the word "null" was perhaps not sufficient as it is IMO sometimes misused.

Patrice

"Scott M." <s-mar@xxxxxxxxxxxxxx> a écrit dans le message de news: OTi6AT4dHHA.1244@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

Hmmm, again, to my knowledge the concepts of "undefined", "unknown", "not applicable" have never been identified with "null" in SQL or anywhere else. When I mark a field in SQL as "nullable", I am not doing it because I don't know what value the user may want to put into the field. On the contrary, I do it because I'm letting the user not put any value at all into the field.

Again "undefined", "unknown", "not applicable" does not equal "null". I think this is super-important to be clear on, because "null" has a special purpose and meaning and the minute you try to attach a meaning that is something similar, but not the same, to it, you confuse the issue and make it more complicated than it need be.

"Patrice" <<http://www.chez.com/scribe/>> wrote in message news:%23wpuDg1dHHA.1868@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

OK I perhaps see a bit better your semantic point. AFAIK some additional "markers" were suggested in SQL standards to better define various meanings of "undefined",

Re: C# Nullable types

"unknown", "not applicable" or several other semantics given to NULL values but never made into products...

I perhaps miss also some nuances as English is not my native language but hopefully the overall picture should be clear enough for the OP.

"Scott M." <s-mar@xxxxxxxxxxxxxx> a écrit dans le message de news: ulQ8h4udHHA.284@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

Something having an unknown value is not, in any way, related to a discussion of null. The very word you use "unknown" implies that there is, in fact, something to "know". A null value indicates the exact opposite of that, that there is no value at all. That's why I said that your analogy was not a good one.

When I run into situations where a value is unknown to me, I set up a variable to capture that value. After doing that, I can then look to see if the variable is null, zero, "green" or anything else. The fact that I didn't know the value of the variable does not imply null.

You seem to be discussing what a "nullable type" is, rather than the meaning of "null".

"Patrice"

Re: C# Nullable types

<<http://www.chez.com/scribe/>>

wrote in message

news:esbV%23RudHHA.4188@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Not sure
what you
meant. How
would you
mark that
the price is
unknown
if not using
a "nullable"
type
representing
a null value
? As a side
note, an
empty
string is not
a "null"
string (in
the first
case we
known that
the value is
a zero
length
string, in
the other
case we
don't know
what the
value is).

IMO one of
the problem
in
discussing
is that null
has multiple
acceptances.
In the
context of a
nullable
type this is
the same
than
the "NULL"
(I'll use
uppercase

Re: C# Nullable types

for this
meaning)
marker used
in most
DB, not the
"null"
reference
as usual in
C# (likely
why MS
used
HasValue
for what is
called
"nullable"
types to a
void the
ambiguity).
VB.NET
uses the
Nothing
keyword.

Finally I see
sometimes
what is
IMO an
abusive use
of NULL. If
you
know that
you have no
name suffix,
you don't
have to use
a NULL
value
but an
empty
string will
do. You
have to use
a
Nullable
column if
you
want to
distinguish
if the value
is an empty
string or if

Re: C# Nullable types

it has no
meaning
(i.e. not
"known",
"applicable"
or whatever
semantic
you
attribute to
the NULL
value).

Patrice

"Scott M."
<s-mar@xxxxxxxxxxxxxx>
a écrit dans
le message
de news:
%23NBysutdHHA.4636@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

I
would
have
to
disagree
with
your
analogy.

If
something
is
free,
it
still
has
a
value
of
zero
dollars
(as
you
say),
but
if
we
don't
know

Re: C# Nullable types

the
price,
that
doesn't
make
the
price
null,
it
just
makes
the
price
unkown.

null
is
simply
a
keyword
that
indicates
that
the
item
in
question
does
not
have
a
relationship
to
any
data
at
all.

A
=
0
<--
A
has
a
value
of
zero
A
=
"

Re: C# Nullable types

"

<--

A

has

a

value

of

the

space

char

A

=

""

<--

A

has

a

null

value

(no

data

at

all)

The

benefit

of

null

values

is

primarily

when

using

databases,

since

most

databases

have

tables

where

not

all

fields

are

required

to

have

a

value

(like

a

Re: C# Nullable types

middle
name
or
apartment
number
or
name
suffix,
such
as
Sr.
or
Jr.).
Since
it
is
possible
that
a
field
may
be
null,
we
need
a
way
of
checking
it
as
such
or
passing
null
values
into
it.

-Scott

"Patrice"

<<http://www.chez.com/scribe/>>

wrote

in

message

news:uiUEhoqdHHA.1220@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

For
example
for

Re: C# Nullable types

a
price
it
would
allow
to
distinguish
between
something
that
is
free
(0)
and
something
for
which
don't
know
the
price
(null).

It's
likely
you'll
mostly
use
this
for
dates
(if
the
vetn
occured
the
date
will
be
filled,
if
the
event
didn't
occured
date,
the
date
will
be
null).

Re: C# Nullable types

Patrice

"AVL"

<AVL@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>

a

écrit

dans

le

message

de

news:

49380D43-FF22-4652-98B0-82E7C30DEC82@xx

hi,

i'm

a

new

bie

to

c#.net

2.0....

i've

come

across

a

new

feature

by

name

nullable

types...need

some

info

on

it

what

is

actually

a

null

value...

what

exactly

is

advantage

we

get

by

Re: C# Nullable types

specifying
a
value
type
as
a
null
value...

please
clarify