Subject: Globally Unique IDs (GUIDs)
Posted by Nils Poldrack on Mon, 05 Apr 2004 09:00:01 GMT
View Forum Message <> Reply to Message

Hello,

Joachim Buechse from Ergon (CH) suggested on 2003-09-25 to use globally
unique
identifier for railML elements. He suggested to use the IP of the
creating computer concatenated with the milliseconds (e.g. since
1970-01-01).

In my "Instructions how to build a railML scheme" I adopted this proposal.
IPv4 and IPv6 create different spaces for their GUIDs.
Advantages of Joachim's proposal are:
- IP is easy to determine.
- The milliseconds since 1970-01-01 are easy to determine.
- There is not force to convert IPv4 to IPv6 or to pad the length.
- The saving format doesn't matter: HEX or Base64.
Disadvantages of Joachim's proposal are:
- No anonymity: via static IP it's possible to find the creator of the GUID.
- Not every computer has got an IP.

Ulrich Linder made the suggestion to use the MAC instead of the IP:
there is anonymity and (almost) every PC has a network card and so a
MAC. (I don't know how to read the MAC - but this is a problem of the
software engineer.)

Please write me your opinion. I prefer Joachim's GUID but using MAC
instead of IP. Thank you there much for your statement.

Nils Poldrack
IVI Dresden

PS. Sorry for my "rusty" English.

Subject: Re: Globally Unique IDs (GUIDs)
Posted by Joerg von Lingen on Tue, 20 Apr 2004 06:35:43 GMT
View Forum Message <> Reply to Message

Hello,

with regard to rolling stock there is no need to invent new GUIDs. Every railway operator use a numbering system for his rolling stock to indentify each single vehicle. Thus it would be simple to use the operator abbreviation, eg. SBB for Swiss Federal Railway, plus the operators vehicle number.

The advantage would be, that such GUIDs are human readable - means everyone could understand what an
identifier like "DB 425 201-1" stands for.

With regard to IP or MAC address: up-to-date desktop computers are normally fitted with a network
interface, but there are units without. And, as far as I know, even the MAC address can be manipulated by software.
--
Regards,
Joerg von Lingen
Institut fuer Bahntechnik GmbH, Dresden

Nils Poldrack wrote the following on 05.04.2004 11:00:

> Hello,
>
> Joachim Buechse from Ergon (CH) suggested on 2003-09-25 to use globally
> unique
> identifier for railML elements. He suggested to use the IP of the
> creating computer concatenated with the milliseconds (e.g. since
> 1970-01-01).
>
> In my "Instructions how to build a railML scheme" I adopted this proposal.
> IPv4 and IPv6 create different spaces for their GUIDs.
> Advantages of Joachim's proposal are:
> - IP is easy to determine.
> - The milliseconds since 1970-01-01 are easy to determine.
> - There is not force to convert IPv4 to IPv6 or to pad the length.
> - The saving format doesn't matter: HEX or Base64.
> Disadvantages of Joachim's proposal are:
> - No anonymity: via static IP it's possible to find the creator of the GUID.
> - Not every computer has got an IP.
>
> Ulrich Linder made the suggestion to use the MAC instead of the IP:
> there is anonymity and (almost) every PC has a network card and so a
> MAC. (I don't know how to read the MAC - but this is a problem of the
> software engineer.)
>
> Please write me your opinion. I prefer Joachim's GUID but using MAC
> instead of IP. Thank you there much for your statement.
>
> Nils Poldrack
> IVI Dresden
>
> PS. Sorry for my "rusty" English.

Subject: Re: Globally Unique IDs (GUIDs)
Posted by thomas.kauer on Thu, 22 Apr 2004 06:06:17 GMT
View Forum Message <> Reply to Message

Hello,

at first sight I would agree with you. For simply being sure not to use
the same number twice (at least within one file) it should work well.

But I see an other possible problem that won't be solved by neither of the
proposed numbering methods since it asks for the reuse of a number given
once:

If the rollingstock schema shall be used to follow the lifecycle of a
vehicle this would make some problems. The vehicles don't remain
unchanged. They may be sold or rent to an other owner or the owner applies
some renewing work on it and since the official numbering includes
describing parts the vehicle will receive a new number and even if it is
only a for a different use of exactly the some vehicle the owner may apply
a new number (for example "DB 261" was set to "DB 361" so that other staff
could be used on it).
Further more the same number may be reused when it's first "owner" has
been sold or destroyed.
So if the goal is to follow a vehicle over different changes applied to it
or to be sure over long term not to get confused with the numbers that
will be reused you better apply an independent unique number that contains
no description - and reuse it whenever giving data for that vehicle...
But how to make sure that the same unique ID will be reused whenever the
same vehicle is mentioned. At the moment given I don't see a good solution
for that.
Perhaps we could add an information like "previous GUID" to the schema so
that a chain could be build to follow the changes when the schema is used
for such purpose?

With regard to IP or MAC address: even if it's not perfect I can't find
better up to now.

Best regards
Thomas Kauer

Joerg von Lingen wrote:

> Hello,

> with regard to rolling stock there is no need to invent new GUIDs. Every

railway operator use a
> numbering system for his rolling stock to indentify each single vehicle.
Thus it would be simple to
> use the operator abbreviation, eg. SBB for Swiss Federal Railway, plus the
operators vehicle number.

> The advantage would be, that such GUIDs are human readable - means everyone
could understand what an
> identifier like "DB 425 201-1" stands for.

> With regard to IP or MAC address: up-to-date desktop computers are normally
fitted with a network
> interface, but there are units without. And, as far as I know, even the MAC
address can be
> manipulated by software.
> --
> Regards,
> Joerg von Lingen
> Institut fuer Bahntechnik GmbH, Dresden

> Nils Poldrack wrote the following on 05.04.2004 11:00:

>> Hello,
>>
>> Joachim Buechse from Ergon (CH) suggested on 2003-09-25 to use globally
>> unique
>> identifier for railML elements. He suggested to use the IP of the
>> creating computer concatenated with the milliseconds (e.g. since
>> 1970-01-01).
>>
>> In my "Instructions how to build a railML scheme" I adopted this proposal.
>> IPv4 and IPv6 create different spaces for their GUIDs.
>> Advantages of Joachim's proposal are:
>> - IP is easy to determine.
>> - The milliseconds since 1970-01-01 are easy to determine.
>> - There is not force to convert IPv4 to IPv6 or to pad the length.
>> - The saving format doesn't matter: HEX or Base64.
>> Disadvantages of Joachim's proposal are:
>> - No anonymity: via static IP it's possible to find the creator of the
GUID.
>> - Not every computer has got an IP.
>>
>> Ulrich Linder made the suggestion to use the MAC instead of the IP:
>> there is anonymity and (almost) every PC has a network card and so a
>> MAC. (I don't know how to read the MAC - but this is a problem of the
>> software engineer.)
>>
>> Please write me your opinion. I prefer Joachim's GUID but using MAC

>> instead of IP. Thank you there much for your statement.
>>
>> Nils Poldrack
>> IVI Dresden
>>
>> PS. Sorry for my "rusty" English.

Subject: Re: Globally Unique IDs (GUIDs)
Posted by thomas.kauer on Thu, 22 Apr 2004 06:06:43 GMT

View Forum Message <> Reply to Message

Hello,

at first sight I would agree with you. For simply being sure not to use the same number twice (at least within one file) it should work well.

But I see an other possible problem that won't be solved by neither of the proposed numbering methods since it asks for the reuse of a number given once:

If the rollingstock schema shall be used to follow the lifecycle of a vehicle this would make some problems. The vehicles don't remain unchanged. They may be sold or rent to an other owner or the owner applies some renewing work on it and since the official numbering includes describing parts the vehicle will receive a new number and even if it is only a for a different use of exactly the some vehicle the owner may apply a new number (for example "DB 261" was set to "DB 361" so that other staff could be used on it).
Further more the same number may be reused when it's first "owner" has been sold or destroyed.
So if the goal is to follow a vehicle over different changes applied to it or to be sure over long term not to get confused with the numbers that will be reused you better apply an independent unique number that contains no description - and reuse it whenever giving data for that vehicle...
But how to make sure that the same unique ID will be reused whenever the same vehicle is mentioned. At the moment given I don't see a good solution for that.
Perhaps we could add an information like "previous GUID" to the schema so that a chain could be build to follow the changes when the schema is used for such purpose?

With regard to IP or MAC address: even if it's not perfect I can't find better up to now.

Best regards
Thomas Kauer

Joerg von Lingen wrote:

> Hello,

> with regard to rolling stock there is no need to invent new GUIDs. Every
railway operator use a
> numbering system for his rolling stock to indentify each single vehicle.
Thus it would be simple to
> use the operator abbreviation, eg. SBB for Swiss Federal Railway, plus the
operators vehicle number.

> The advantage would be, that such GUIDs are human readable - means everyone
could understand what an
> identifier like "DB 425 201-1" stands for.

> With regard to IP or MAC address: up-to-date desktop computers are normally
fitted with a network
> interface, but there are units without. And, as far as I know, even the MAC
address can be
> manipulated by software.
> --
> Regards,
> Joerg von Lingen
> Institut fuer Bahntechnik GmbH, Dresden

> Nils Poldrack wrote the following on 05.04.2004 11:00:

>> Hello,
>>
>> Joachim Buechse from Ergon (CH) suggested on 2003-09-25 to use globally
>> unique
>> identifier for railML elements. He suggested to use the IP of the
>> creating computer concatenated with the milliseconds (e.g. since
>> 1970-01-01).
>>
>> In my "Instructions how to build a railML scheme" I adopted this proposal.
>> IPv4 and IPv6 create different spaces for their GUIDs.
>> Advantages of Joachim's proposal are:
>> - IP is easy to determine.
>> - The milliseconds since 1970-01-01 are easy to determine.
>> - There is not force to convert IPv4 to IPv6 or to pad the length.
>> - The saving format doesn't matter: HEX or Base64.
>> Disadvantages of Joachim's proposal are:
>> - No anonymity: via static IP it's possible to find the creator of the

GUID.
>> - Not every computer has got an IP.
>>
>> Ulrich Linder made the suggestion to use the MAC instead of the IP:
>> there is anonymity and (almost) every PC has a network card and so a
>> MAC. (I don't know how to read the MAC - but this is a problem of the
>> software engineer.)
>>
>> Please write me your opinion. I prefer Joachim's GUID but using MAC
>> instead of IP. Thank you there much for your statement.
>>
>> Nils Poldrack
>> IVI Dresden
>>
>> PS. Sorry for my "rusty" English.

Subject: Re: Globally Unique IDs (GUIDs)
Posted by Wolfgang Keller on Tue, 18 May 2004 12:04:49 GMT

View Forum Message <> Reply to Message

Hello,

sorry for bursting into the group just like this, but...

Am Mon, 05 Apr 2004 11:00:01 +0200 schrieb Nils Poldrack:

> Joachim Buechse from Ergon (CH) suggested on 2003-09-25 to use globally
> unique
> identifier for railML elements. He suggested to use the IP of the
> creating computer concatenated with the milliseconds (e.g. since
> 1970-01-01).

Such a GUID schema would not fulfil one essential requirement: It would not allow to exchange data edited by different people on different computers at different moments with different software in a consistent way (therefore, such a GUID would be essentially useless).

To achieve this, a GUID must be defined in such a way that the same physical "real world" object gets always the same GUID, no matter who edits the set of data representing the object in the computer system, when and with what software.

In other standardisation groups working on data models of any kind where the data represents physical objects in the "real world" this requirement is usually respected.

In maintenance software, for example, maintenance assets are identified

within the computer system by the ID of the manufacturer of the asset plus
the serial number of the asset as it is indicated on the identification
plate.

Best regards,

Wolfgang Keller

Subject: Re: Globally Unique IDs (GUIDs)
Posted by thomas.kauer on Fri, 28 May 2004 10:22:31 GMT
View Forum Message <> Reply to Message

Hello,

at first sight I would agree with you.
But this needs an global definition of the "real" object. When is it still
the same object and when does it change it's ID!?
For example in an application for the needs of track maintenance you will
probably change the internal ID of an object (track, switch, ...) when
after a rework the object has moved a little bit. For them they got 2
objects at 2 different positions and 2 different times. For most other
users this is still the same object with just changed attributes.

And if you agree on when to change an ID - what could build the ID? What
is the manufacturer of a track or switch and what the serial number of the
asset? What is to use when you are designing new tracks that are only used
for studies and never will be constructed? What else should be used?

Most systems I know use a combination of different attributes like
"station name" and "switch name" to identify the objects they exchange
with other systems (most of them also use an arbitrary ID that is only
valid within the application or DB system). To integrate external data
they have to apply these attributes to match the elements with those they
already have - if existing.
So for the integration of the data from a RailML data file the use of
attributes will be the choice for most of the systems. The ID's defined in
RailML can be used to get the connected/referenced data like visualisation
of an element within one RailML data set - they are consistent and unique
and easy to use.

If you need to have different states of the same object in your system you
have to add information to make this difference. For that you might need a
complex status attribute.
To make the match after a changing in the key attributes set is more
complicated - you need information about the former values of the
attributes.

For me this subject is not solved in a global way yet and needs more
discussion.

Best regards,
Thomas Kauer

Wolfgang Keller wrote:

> Hello,

> sorry for bursting into the group just like this, but...

> Am Mon, 05 Apr 2004 11:00:01 +0200 schrieb Nils Poldrack:

>> Joachim Buechse from Ergon (CH) suggested on 2003-09-25 to use globally
>> unique
>> identifier for railML elements. He suggested to use the IP of the
>> creating computer concatenated with the milliseconds (e.g. since
>> 1970-01-01).

> Such a GUID schema would not fulfil one essential requirement: It would not
> allow to exchange data edited by different people on different computers at
> different moments with different software in a consistent way (therefore,
> such a GUID would be essentially useless).

> To achieve this, a GUID must be defined in such a way that the same
> physical "real world" object gets always the same GUID, no matter who edits
> the set of data representing the object in the computer system, when and
> with what software.

> In other standardisation groups working on data models of any kind where
> the data represents physical objects in the "real world" this requirement
> is usually respected.

> In maintenance software, for example, maintenance assets are identified
> within the computer system by the ID of the manufacturer of the asset plus
> the serial number of the asset as it is indicated on the identification
> plate.

> Best regards,

> Wolfgang Keller