

作成日: 2013/03/08		SQL Server 2008 R2				PostgreSQL 9.2x			
参考URI		<a href="http://msdn.microsoft.com/ja-jp/library/ms187594%28SQL100%29.aspx">http://msdn.microsoft.com/ja-jp/library/ms187594%28SQL100%29.aspx</a>				参考URI			
参考URI		<a href="http://www.postgresql.jp/document/9.2/html/datatype.html">http://www.postgresql.jp/document/9.2/html/datatype.html</a>				参考URI			
数値									
No	用途	型名	ISO名	格納サイズ	範囲	変換指針特記 (記載なしは特に考慮不要)	型名	格納サイズ	範囲
1	小さな整数	<b>tinyint</b>		1バイト	0 ~ 255				
2	狭範囲整数	<b>smallint</b>		2バイト	-2 <sup>15</sup> (-32,768) ~ 2 <sup>15</sup> -1 (32,767)		<b>smallint/int2</b>	2バイト	-32768から+32767
3	標準的整数	<b>int</b>	integer	4バイト	-2 <sup>31</sup> (-2,147,483,648) ~ 2 <sup>31</sup> -1 (2,147,483,647)		<b>integer/int4</b>	4バイト	-2147483648から+2147483647
4	広範囲整数	<b>bigint</b>		8バイト	-2 <sup>63</sup> (-9,223,372,036,854,775,808) ~ 2 <sup>63</sup> -1 (9,223,372,036,854,775,807)		<b>bigint/int8</b>	8バイト	-9223372036854775808から9223372036854775807
5	自動採番の仕組み	(属性) <b>IDENTITY(s,i)</b> を数値型に付与				精度によって、smallserial, serial, bigserialを選択 既存値最大を初期値をリセットする必要がある	<b>smallserial/serial2</b> <b>serial</b> <b>bigserial/serial8</b>	2バイト 4バイト 8バイト	1から32767 1から2147483647 1から9223372036854775807
概数									
No	用途	型名	ISO名	格納サイズ	範囲	変換指針特記 (記載なしは特に考慮不要)	型名	格納サイズ	範囲
1	ユーザ指定浮動小数値	<b>decimal(p,s)</b>	Dec	1-9桁: 5バイト 10-19桁: 9バイト 20-28桁: 13バイト 29-38桁: 17バイト	-10 <sup>38</sup> +1 ~ 10 <sup>38</sup> -1		<b>decimal</b>	可変長	小数点前までは131072桁, 小数点以降は16383桁
2	ユーザ指定浮動小数値	<b>numeric(p,s)</b>	Dec	1-9桁: 5バイト 10-19桁: 9バイト 20-28桁: 13バイト 29-38桁: 17バイト	-10 <sup>38</sup> +1 ~ 10 <sup>38</sup> -1		<b>numeric</b>	可変長	小数点前までは131072桁, 小数点以降は16383桁
3	浮動小数点数値	<b>real</b>	real	4バイト	-3.40E+38 ~ -1.18E-38, 0, および 1.18E-38 ~ 3.40E+38		<b>real/float4</b>	4バイト	6桁精度
4	浮動小数点数値	<b>float(n)</b>	float Double precision	1-24桁: 4バイト 23-53桁: 8バイト	-1.79E+308 ~ -2.23E-308, 0, および 2.23E-308 ~ 1.79E+308		<b>double precision/float8</b>	8バイト	15桁精度
通貨									
No	用途	型名	ISO名	格納サイズ	範囲	変換指針特記 (記載なしは特に考慮不要)	型名	格納サイズ	範囲
1	通貨型	<b>money</b>		8バイト	-922,337,203,685,477.5808 ~ 922,337,203,685,477.5807		<b>money</b>	8バイト	-92233720368547758.08 から +92233720368547758.07
2	通貨型(少額)	<b>smallmoney</b>		4バイト	-214,748.3648 ~ 214,748.3647				
文字									
No	用途	型名	ISO名	格納サイズ	範囲	変換指針特記 (記載なしは特に考慮不要)	型名	格納サイズ	範囲
1	可変長文字列	<b>varchar(n)</b>	character varying	2nバイト 最大n:8000	8000文字		<b>varchar(n)</b>	126バイトまで:n文字+1バイト~最大1ギガバイト 126バイト以上:n文字+4バイト~最大1ギガバイト	
2	Unicode(UCS2) 可変長文字列	<b>nvarchar(n)</b>	national character varying	2+2nバイト 最大n:4000	4000文字	文字セットをUNICODE(UTF-16)にする必要がある。			
3	固定長文字列	<b>char(n)</b>	character(n)	nバイト 最大n:8000	8000文字		<b>char(n)</b>	1バイト 126バイトまで:n文字+1バイト~最大1ギガバイト 126バイト以上:n文字+4バイト~最大1ギガバイト	
4	Unicode(UCS2) 固定長文字列	<b>nchar(n)</b>	national character(n)	2nバイト 最大n:4000	4000文字	文字セットをUNICODE(UTF-16)にする必要がある。			
5	文字列 将来のバージョンで削除予定	<b>text</b>		最大 2,147,483,647バイト	2 <sup>31</sup> -1 (2,147,483,647)		<b>text</b>		
6	Unicode文字列 将来のバージョンで削除予定	<b>ntext</b>	national text	最大 1073741823 * 2バイト	2 <sup>30</sup> -1 (1,073,741,823)				
バイナリデータ									
No	用途	型名	ISO名	格納サイズ	範囲	変換指針特記 (記載なしは特に考慮不要)	型名	格納サイズ	範囲
1	可変長バイナリ	<b>varbinary (n)</b>	Binary varying	2バイト + nバイト (最大8000)	1~8000				
2	一意の 2 進数を公開するデータ型 (時間関連の内部データ型)	<b>rowversion</b>	timestamp (非SQL2003)	8バイト					
3	固定長バイナリ	<b>binary</b>		nバイト (最大 8000)	1~8000		<b>bytea</b>	1または4バイトと実際のバイナリ列の長さ	1Gバイト
4	可変長バイナリ	<b>image</b>		最大 2,147,483,647バイト	0 ~ 2 <sup>31</sup> -1 (2,147,483,647)	1GBを超える場合は、ラージオブジェクトを使用、 外部ファイルに保存、複製カラム化等を検討			
日付/時刻									
No	用途	型名	ISO名	格納サイズ	範囲	変換指針特記 (記載なしは特に考慮不要)	型名	格納サイズ	範囲
1	24 時間形式の時刻と組み合わせた日付	<b>datetime</b>		8 バイト	日付範囲: 1753 年 1 月 1 日 ~ 9999 年 12 月 31 日 時刻範囲: 00:00:00 ~ 23:59:59.997	文字列表現を用いて移行 文字列表現等詳細: <a href="http://msdn.microsoft.com/ja-jp/library/bb630352%28v=sql.100%29.aspx">http://msdn.microsoft.com/ja-jp/library/bb630352%28v=sql.100%29.aspx</a>			
2	24 時間形式の時刻と組み合わせた日付 (1秒以下指定桁)	<b>datetime2(p)</b>		1秒以下3未満 6バイト 1秒以下3~4 7バイト 1秒以下5以上 8バイト	日付範囲: 0001-01-01 ~ 9999-12-31 西暦 1 年 1 月 1 日 ~ 西暦 9999 年 12 月 31 日 時刻範囲: 00:00:00 ~ 23:59:59.9999999	文字列表現を用いて移行 ナノ秒が必要な場合は、変換できない(桁落ち) →数値・文字列表現等でアプリケーションでなんらかの対応が必要	<b>timestamp(p) [without time zone]</b>	8バイト	4713 BC から 294276 AD
3	24 時間形式の時刻と組み合わせた日付 (1秒未満は保持しない)	<b>smalldatetime</b>		4バイト	日付範囲: 1900-01-01 ~ 2079-06-06 1900 年 1 月 1 日 ~ 2079 年 6 月 6 日 時刻範囲: 00:00:00 ~ 23:59:59	文字列表現を用いて移行			
4	タイムゾーンを認識する 24 時間形式の時刻と組み合わせた日付	<b>datetimeoffset(p)</b>		datetimeoffset : 10バイト 1秒以下 7桁 datetimeoffset(0): 8バイト 1秒以下 0-2桁 datetimeoffset(1): 8バイト 1秒以下 0-2桁 datetimeoffset(2): 8バイト 1秒以下 0-2桁 datetimeoffset(3): 9バイト 1秒以下 3-4桁 datetimeoffset(4): 9バイト 1秒以下 3-4桁 datetimeoffset(5): 10バイト 1秒以下 5-7桁 datetimeoffset(6): 10バイト 1秒以下 5-7桁 datetimeoffset(7): 10バイト 1秒以下 5-7桁	日付範囲: 0001-01-01 ~ 9999-12-31 西暦 1 年 1 月 1 日 ~ 西暦 9999 年 12 月 31 日 時刻範囲: 00:00:00 ~ 23:59:59.9999999	文字列表現を用いて移行 ナノ秒が必要な場合は、変換できない(桁落ち) →数値・文字列表現等でアプリケーションでなんらかの対応が必要	<b>timestamp(p) with time zone</b>	8バイト	4713 BC から 294276 AD

作成日: 2013/03/08		SQL Server 2008 R2				PostgreSQL 9.2x			
		参考URI	http://msdn.microsoft.com/ja-jp/library/ms187594%28SQL-100%29.aspx			参考URI	http://www.postgresql.jp/document/9.2/html/datatype.html		
5	日付	date		3バイト	0001-01-01 ~ 9999-12-31 西暦 1 年 1 月 1 日 ~ 西暦 9999 年 12 月 31 日 (文字列表現 10 文字)	文字列表現を用いて移行	date	4バイト	4713 BC から 5874897 AD
6	時刻	time(p)		time : 5バイト 1秒以下 7桁 time(0): 3バイト 1秒以下 0-2桁 time(1): 3バイト 1秒以下 0-2桁 time(2): 3バイト 1秒以下 0-2桁 time(3): 4バイト 1秒以下 3-4桁 time(4): 4バイト 1秒以下 3-4桁 time(5): 5バイト 1秒以下 5-7桁 time(6): 5バイト 1秒以下 5-7桁 time(7): 5バイト 1秒以下 5-7桁	00:00:00.0000000 ~ 23:59:59.9999999	文字列表現を用いて移行 ナ/桁が必要な場合は、変換できない(桁落ち) → 数値・文字列表現等でアプリケーションでなんとかするなど必要	time[(p)] [without time zone]	8バイト	00:00:00 から 24:00:00
ビット									
No	用途	型名	ISO名	格納サイズ	範囲	変換指針特記 (記載なしは特に考慮不要)	型名	格納サイズ	範囲
1	1, 0, または NULL の値をとる整数型	bit		1バイト(8ビット毎)	1, 0, または NULL (TRUE:1/FALSE:0に変換する)		bit(n)	1バイト(8ビット毎) (ビット列長に応じた5または8バイトのオーバーヘッド)	1, 0, NULL
その他									
No	用途	型名	ISO名	格納サイズ	範囲	変換指針特記 (記載なしは特に考慮不要)	型名	格納サイズ	範囲
1	XML	xml ([CONTENT   DOCUMENT] Xml_schema_collection)		2Gバイト	-	文字列表現を用いて、移行する # varchar/mvarcharに変換可能 # http://msdn.microsoft.com/ja-jp/library/ms189887%28v=sql.100% 操作に関しては、 別途アプリケーション仕様(Transact-SQLで操作していたため)を検討し、 PostgreSQLの操作関数等を利用を検討する必要がある。	xml	記載なし	-
2	GUID	uniqueidentifier		16バイト	xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx (16進)		UUID	16バイト	xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx (16進)
その他 (Microsoft SQL Serverユニーク)									
No	用途	型名	ISO名	格納サイズ	範囲	変換指針	型名	格納サイズ	範囲
1	ストアドプロシージャ用の結果等保持用	cursor		-	NULL許可	別途アプリケーション仕様を検討	-	-	-
2	階層内の位置 SQLサーバテーブル階層構造走査時の保持用	hierarchyid		-	-	別途アプリケーション仕様を検討	-	-	-
3	ユーザー定義 (共通言語ランタイム)	CLR		-	-	別途アプリケーション仕様を検討	-	-	-
4	ユーザー定義関数の列、 パラメータ、変数、および戻り値	sql_variant		-	-	別途アプリケーション仕様を検討	-	-	-
5	行のセットの一時的な格納場所	table		-	-	別途アプリケーション仕様を検討	-	-	-

作成日: 2014/03/06		SQL Server 2005				PostgreSQL 9.2				
参考URL		http://msdn.microsoft.com/ja-jp/library/ms187752(v=sql90).aspx 等				参考URL				
http://www.postgresql.jp/document/9.2/html/datatype.html 等										
文字列										
No	用途	型名	ISO名	格納サイズ	範囲	変換時の留意点(記載なしは考慮不要)	型名	別名	格納サイズ	範囲
1	固定長文字列	CHAR [ ( n ) ] n省略:16(1) or 30(*2) n範囲:1~8000	CHARACTER [ ( n ) ]	固定長(space埋め) nバイト	8000バイト分の文字		CHARACTER [ ( n ) ] n省略:1	CHAR [ ( n ) ]	固定長(space埋め) 128バイトまで: n * 文字のバイト数 + 1バイト 128バイト以上: n * 文字のバイト数 + 4バイト	
2	可変長文字列	VARCHAR [ ( n ) ] n省略:16(1) or 30(*2) n範囲:1~8000	CHAR VARYING [ ( n ) ] CHARACTER VARYING [ ( n ) ]	可変長 n * 2 バイト	8000バイト分の文字	SQL Server側では、Unicode以外の文字コードはデータ長nをバイト単位で指定しているが、PostgreSQL側では文字数単位で指定する必要がある。	CHARACTER VARYING ( n )	VARCHAR ( n )	可変長 128バイトまで: n * 文字のバイト数 + 1バイト 128バイト以上: n文字 + 4バイト	
3	上限の無い可変長文字列	VARCHAR ( max )	CHAR VARYING ( max ) CHARACTER VARYING ( max )	可変長 最大2^31-1バイト ※使用する文字列によりサイズ変動	2^31-1 (2,147,483,647)バイト分の文字		CHARACTER VARYING n省略:制限無し	VARCHAR	可変長 最大1ギガバイト	約1GB分の文字
4	上限の無い可変長文字列 (将来のバージョンで削除予定で、 VARCHAR ( max )の使用を推奨)	TEXT	-	可変長 最大2^31-1バイト ※使用する文字列によりサイズ変動	2^31-1 (2,147,483,647)バイト分の文字	SQL Server側でデータサイズが1GB以上のデータは、PostgreSQL側の対応するデータ型には格納できないため、ラージオブジェクト、または外部ファイルの使用を検討する必要がある。	TEXT	-	可変長 最大1ギガバイト	約1GB分の文字
Unicode文字列										
No	用途	型名	ISO名	格納サイズ	範囲	変換時の留意点(記載なしは考慮不要)	型名	別名	格納サイズ	範囲
1	Unicode(UCS2) 固定長文字列	NCHAR [ ( n ) ] n省略:16(1) or 30(*2) n範囲:1~4000	NATIONAL CHAR [ ( n ) ] NATIONAL CHARACTER [ ( n ) ]	固定長(space埋め) n * 2バイト	Unicode(UCS-2)で4000文字		CHARACTER [ ( n ) ] n省略:1	CHAR [ ( n ) ]	固定長(space埋め) 128バイトまで: n * 文字のバイト数 + 1バイト 128バイト以上: n * 文字のバイト数 + 4バイト	
2	Unicode(UCS2) 可変長文字列	NVARCHAR [ ( n ) ] n省略:16(1) or 30(*2) n範囲:1~4000	NATIONAL CHAR VARYING [ ( n ) ] NATIONAL CHARACTER VARYING [ ( n ) ]	可変長 n * 2 * 2 バイト	Unicode(UCS-2)で4000文字		CHARACTER VARYING ( n )	VARCHAR ( n )	可変長 128バイトまで: n * 文字のバイト数 + 1バイト 128バイト以上: n文字 + 4バイト	
3	Unicode(UCS2) 上限の無い可変長文字列	NVARCHAR ( max )	NATIONAL CHAR VARYING ( max ) NATIONAL CHARACTER VARYING ( max )	可変長 最大2^31-1バイト	Unicode(UCS-2)で2^30-1 (0,073,741,823)文字		CHARACTER VARYING	VARCHAR	可変長 最大1ギガバイト	約1GB分の文字
4	Unicode(UCS2) 上限の無い可変長文字列 (将来のバージョンで削除予定で、 NVARCHAR ( max )の使用を推奨)	NTEXT	NATIONAL TEXT	可変長 最大2^31-1バイト	Unicode(UCS-2)で2^30-1 (0,073,741,823)文字	SQL Server側でデータサイズが1GB以上のデータは、PostgreSQL側の対応するデータ型には格納できないため、ラージオブジェクト、または外部ファイルの使用を検討する必要がある。	TEXT	-	可変長 最大1ギガバイト	約1GB分の文字
真数										
No	用途	型名	ISO名	格納サイズ	範囲	変換時の留意点(記載なしは考慮不要)	型名	別名	格納サイズ	範囲
1	固定長の有効桁数と、 小数点以下桁数を持つ任意精度の数値	DECIMAL ( p , s ) NUMERIC ( p , s ) n範囲:有効桁1~38桁 s範囲:小数点以降0~p桁	DEC ( p , s )	1~9桁: 5バイト 10~19桁: 9バイト 20~28桁: 13バイト 29~38桁: 17バイト	- 10^38 + 1 ~ 10^38 - 1		NUMERIC ( p , s ) p範囲:有効桁1~1000桁 s範囲:小数点以降0~p桁	DECIMAL ( p , s )		- 10^1000 + 1 ~ 10^1000 - 1
2		DECIMAL ( p ) NUMERIC ( p ) s省略:小数点以降0桁 n範囲:有効桁1~38桁	DEC ( p )	1~9桁: 5バイト 10~19桁: 9バイト 20~28桁: 13バイト 29~38桁: 17バイト	- 10^38 + 1 ~ 10^38 - 1		NUMERIC ( p ) p範囲:有効桁1~1000桁	DECIMAL ( p )		- 10^1000 + 1 ~ 10^1000 - 1
3		DECIMAL NUMERIC p省略:有効桁18桁 s省略:小数点以降0桁	DEC	9バイト	- 10^18 + 1 ~ 10^18 - 1	SQL Server側で精度と長さの両方が省略されている場合、PostgreSQL側に移行する際に明確に指定を直す必要がある。	NUMERIC ( 18 , 0 )	DECIMAL ( 18 , 0 )		- 10^18 + 1 ~ 10^18 - 1
4		-	-	-	-	SQL Server側で最大精度(有効桁38桁)を使用している場合、精度がお高い方がよいケースでは、省略による最大精度の使用を検討する必要がある。	NUMERIC	DECIMAL		小数点前までは131072桁 小数点以降は16383桁
5	金銭や通貨の値	MONEY	-	8バイト	-922,337,203,685,477,5808 ~ 922,337,203,685,477,5807		MONEY	-	8バイト	-9223372036854775808 ~ 9223372036854775807
6	少額金銭や通貨の値	SMALLMONEY	-	4バイト	- 214,748,3648 ~ 214,748,3647	SQL Server側のSMALLMONEYからの移行では、桁数が大きくなることにより計算ロジックの見直しが必要になる場合がある。	MONEY	-	8バイト	-9223372036854775808 ~ 9223372036854775807
7	1桁の2進数	BIT	-	※テーブルでは1バイト(8桁)単位で格納する。8列未満の場合、1バイトで格納する。	1, 0, または NULL の値をとる整数型	SQL Server側では、bit型の値を0と1の整数値といだけでなく、True、Falseの文字列としても扱える点に注意が必要である。 PostgreSQL側では、bit型は整数値ではなく0と1のビット文字列であるため、ストアドの移行時にはビット文字列にキャスト等の考慮が必要である。	BIT [ ( n ) ] n省略:1	-	1バイト(8ビット毎) (ビット列長に応じた5または8バイトのオーバーヘッド)	1, 0, またはNULL(NULL許容時)
8	小さな整数	TINYINT	-	1バイト	0 ~ 255	PostgreSQL側に移行する際には、用途によってはテーブルの列定義時に、255以下の制約を整数型smallintでなくともいかに付与することを検討する。また、ストアドの変数宣言でsmallint等の整数型で代用する場合にも別途検討が必要場合がある。	SMALLINT	INT2	2バイト	整数(0 ~ 255)
9	狭範囲整数	SMALLINT	-	2バイト	-2^15 (-32,768) ~ 2^15-1 (32,767)		SMALLINT	INT2	2バイト	整数(-32768 ~ +32767)
10	標準的整数	INT	INTEGER	4バイト	-2^31 (-2,147,483,648) ~ 2^31-1 (2,147,483,647)		INTEGER	INT, INT4	4バイト	整数(-2147483648 ~ -2147483647)
11	広範囲整数	BIGINT	-	8バイト	-2^63 (-9,223,372,036,854,775,808) ~ 2^63-1 (9,223,372,036,854,775,807)		BIGINT	INT8	8バイト	整数(-9223372036854775808 ~ 9223372036854775807)
12	自動採番の仕組み (属性) IDENTITY ( s , 1 ) を整数型に付与	-	-	-	-	SQL Server側における整数型の精度によって、smallserial, serial, bigserialを選択し、また既存値最大の初期値をリセットする必要がある。	SMALLSERIAL SERIAL BIGSERIAL	SERIAL2 SERIAL4 SERIAL8	2バイト 4バイト 8バイト	1から32767 1から2147483647 1から9223372036854775807
概数										
No	用途	型名	ISO名	格納サイズ	範囲	変換時の留意点(記載なしは考慮不要)	型名	別名	格納サイズ	範囲
1	倍精度浮動小数点数	FLOAT	DOUBLE PRECISION	8バイト	- 1.79E+308 ~ -2.23E-308, 0, および 2.23E-308 ~ 1.79E+308		DOUBLE PRECISION	FLOAT8 FLOAT [ ( n ) ] n省略:38桁(2進数) n範囲:25 ~ 53桁(2進数)	8バイト	最低15桁(10進数) およそ-1E+308と+1E+308
2	倍精度浮動小数点数	FLOAT	FLOAT [ ( n ) ] n省略:53桁(2進数) n範囲:25 ~ 53桁(2進数)	8バイト	- 1.79E+308 ~ -2.23E-308, 0, および 2.23E-308 ~ 1.79E+308		DOUBLE PRECISION	FLOAT [ ( n ) ] n省略:38桁(2進数) n範囲:25 ~ 53桁(2進数)	8バイト	最低15桁(10進数) およそ-1E+308と+1E+308
3	単精度浮動小数点数	REAL	FLOAT ( n ) n範囲:1 ~ 24桁(2進数)	4バイト	- 3.40E+38 ~ -1.18E-38, 0, および 1.18E-38 ~ 3.40E+38		REAL	FLOAT4 FLOAT [ ( n ) ] n範囲:1 ~ 24桁(2進数)	4バイト	最低6桁(10進数) 少なくとも-1E+37と+1E+37

作成日: 2014/03/06		SQL Server 2005				PostgreSQL 9.2				
		参考URL		http://msdn.microsoft.com/ja-jp/library/ms187752(v=sql90).aspx 等		参考URL		http://www.postgresql.jp/document/9.2/html/datatype.html 等		
<b>日付と時刻</b>										
No	用途	型名	ISO名	格納サイズ	範囲	変換時の留意点(記載なしは考慮不要)	型名	別名	格納サイズ	範囲
1	日付と時刻	DATETIME	-	8バイト (4+4)	範囲:1753年1月1日～9999年12月31日 精度:3.33秒		TIMESTAMP	-	8バイト	日付と時刻両方(時間帯なし) 11桁, 11桁 4713 BC～254276 AD
2	低精度の日付と時刻	SMALLDATETIME	-	10バイト (9+9)	範囲:1990年1月1日～2079年6月6日 精度:1分	PostgreSQL側に移行する際に、smalldatetimeからTIMESTAMPへは、文字列表現を用いて移行可能である。 ストアドの変数宣言でTIMESTAMPで代用する場合は別途検討が必要な場合がある。				
<b>バイナリ文字列</b>										
No	用途	型名	ISO名	格納サイズ	範囲	変換時の留意点(記載なしは考慮不要)	型名	別名	格納サイズ	範囲
1	固定長バイナリ	BINARY [ ( n ) ] n省略:1 n範囲:1～8000	-	固定長 nバイト	8000バイト分のバイナリ	PostgreSQL側に移行する際に、データ長の指定は不可となり、固定長ではなくなってしまう点に注意が必要である。				
2	可変長バイナリ	VARBINARY [ ( n ) ] n省略:1 n範囲:1～8000	BINARY VARYING [ ( n ) ]	可変長 n+2バイト	8000バイト分のバイナリ	PostgreSQL側に移行する際に、データ長の指定が不可になる点に注意が必要である。				
3	上限の無い可変長バイナリ	VARBINARY ( max )	BINARY VARYING ( max )	可変長 最大2 <sup>31</sup> -1 (2,147,483,647) バイト	2 <sup>31</sup> -1 (2,147,483,647) バイト分のバイナリ		BYTEA	-	可変長 または1バイトと実際のバイナリ列の長さ	1ギガバイト
4	上限の無い可変長バイナリ (将来のバージョンで削除予定で、 VARBINARY ( max )の使用を推奨)	IMAGE	-	可変長 最大2 <sup>31</sup> -1 (2,147,483,647) バイト	2 <sup>31</sup> -1 (2,147,483,647) バイト分のバイナリ	SQL Server側でデータサイズが1GB以上のデータは、PostgreSQL側の対応するデータ型には格納できないため、ラージオブジェクト、または外部ファイルの使用を検討する必要がある。				
<b>その他のデータ型</b>										
No	用途	型名	ISO名	格納サイズ	範囲	変換時の留意点(記載なしは考慮不要)	型名	別名	格納サイズ	範囲
1	ストアドにおける結果保持用のデータ型 カーソルへの参照を格納	CURSOR	-	-	Null許容	カーソルで使用されるデータ型であるため、カーソル機能自体の移行の観点も含め別途検討が必要である。	REFCURSOR	-	-	Null許容
2	ユーザー定義関数の列、 パラメータ、変数、および戻り値用のデータ型	SQL_VARIANT	-	-	text, ntext, image, timestamp, および sql_variant を除き、サポートしている各種 データ型の値が格納可能	別途アプリケーション仕様の検討が必要である。				
3	行のセットの一時的な格納場所となるデータ型	TABLE	-	-	-	対応するデータ型はないが、PostgreSQL側のストアドへの移行では、一時テーブル作成や、SQL 関数等を利用しての対応を検討すると想定される。				
4	一意の2進数を公開するデータ型	TIMESTAMP	ROWVERSION	-	-	SQL Server側からPostgreSQL側に移行する際に、“一意となる番号を格納するバイナリ”としての特性を継続するには別途検討が必要である。	BYTEA	-	入力データサイズ + (または4バイト)	1ギガバイト
5	16バイトのグローバルな一意識別子 (GUID)	UNIQUEIDENTIFIER	-	16バイト	xxxxxxxx-xxxx-xxxx-xxxx- xxxxxxxxxxxx (16進)		UUID	-	16バイト	xxxxxxxx-xxxx-xxxx-xxxx- xxxxxxxxxxxx (16進)
6	XMLドキュメントやそのフラグメントを格納する データ型	XML	-	2ギガバイト	-	XML型の値に対して機能する関数等を含めて別途検討が必要である。	XML	-	-	-

\*1 データ定義または変数宣言ステートメントで指定省略時

\*2 CAST 関数および CONVERT 関数で指定省略時