変更履歴

No	変更日	変更者	変更理由	変更箇所	変更内容	備考
1	2014/04/16	青柳	–	–	新規作成	

Element List

#	Attribute name	Outline	Correspondence element	html	xml	Parameter form	note
1	data-load	It describes, when acquiring initial display data at the time of a display.	Not specified(∗1)	○	○	JSON(Object/Array)	Only the first setting becomes effective when specified for two or more tag.
2	data-func	It describes, when carrying out submit of form, and performing crud or a mail function.	form	○		JSON(Object/Array)	The condition of not depending on the input is specified for a parameter of not data-cond but data-func.
3	data-cond	It uses it together with data-func. It describes in order to give conditions when performing the method in which a condition setup of crud is possible. Although the COND parameter of data-func cannot perform change from a client, data-cond can change conditions according to an input.	input	○		JSON(Object)	It is use that data-func was described by the child element.
4	data-iterator	It describes, when the number of the execution results of crud is plural lines, and repeating an element at arbitrary places.	Not specified	○	○	String	It is necessary to specify bind at the time of specifying "select" or "selectAll" by data-load and method of data-func as the subordinate of an element who specified data-iterator.
5	data-keep	Although deleted by default, the brother element of the element which specified data-iterator is not deleted when this attribute is specified.	Not specified	○	○	–	
6	data-bind-in	It describes, when registering and updating the information on form to DB.	input textarea	○		JSON(Object)	
7	data-bind-out	It describes, when bind the extracted data, a request parameter, etc. at arbitrary places.	Not specified	○	○	JSON(Object/Array)	It is necessary to specify bind at the time of specifying "select" or "selectAll" by data-load and method of data-func as the subordinate of an element who specified data-iterator.
8	data-case	It describes to change and take out a screen according to arbitrary conditions.	Not specified	○	○	JSON(Object/Array)	The plural designation becomes the and condition
9	data-erase	It can be set up not to output the element in which this attribute was specified, or a child element to a screen.	Not specified	○	○	String	
10	data-url	A request URL pattern is set up. URL is determined by the predetermined rule when this attribute is omitted.	Not specified(∗1)	○	○	JSON(Array)	Only the first setting becomes effective when specified for two or more tag.
11	data-auth	When the attestation is necessary for accessing a screen concerned, this attribute is specified.	Not specified(∗1)	○	○	–	
12	data-image	It describes, when acquiring the picture of the picture path (non-Web domain) stored in DB. It is also possible to resize by specifying width and height.	Not specified	○		JSON(Array)	

(∗1)In the case of HTML, it recommends specifying it as a HEAD element.

data-load

*select a value from setting value designated.

search function

JSON(Object)	Parameter Name	Rquired	Setting Value	Description
{				
"TYPE": "crud",	TYPE		crud	It is specified as the case "crud" where a CRUD function is used. (Default)
"METHOD": "select",	METHOD	○	select	It specifies, when extracting data from the table which specified arbitrary search conditions and was set as target.
			selectAll	It specifies, when specifying arbitrary search conditions and extracting all the data including a related table (FK).
			selectByPk	One record of an target table is extracted. data-bind-out can be described, without specifying data-iterator. When this method is specified, it becomes an error when the extraction result is not one.
"TARGET": "shop",	TARGET	○	Table name	It specifies the table name for processing.
"RESULT": "result1",	RESULT	○	Arbitrary character strings	It is an identifier used by access to search results. ※It is assumed a unique name in the page.
"OFFSET": "10",	OFFSET		Arbitrary numerical values (int)	It specifies the position which starts acquisition ※In the case of "select" or "selectAll", "METHOD" is effective ※The line returned is not guaranteed when not specifying "SORT".
"LIMIT": "50",	LIMIT		Arbitrary numerical values (int)	It specifies the acquisition number. ※In the case of "select" or "selectAll", "METHOD" is effective ※The line returned is not guaranteed when not specifying "SORT".
"PAGER": [	PAGER		Object	Pager information object It is specified when displaying a pager. ※The directions for pager information are referring to the [[supplementary] pager specification]. ※In the case of "select" or "selectAll", "METHOD" is effective
"PARAM_NAME": "page",	PARAM_NAME		Arbitrary character strings	It specifies the parameter name of page number. "page" is used when not specifying.
"PERPAGE": "10",	PERPAGE		Arbitrary numerical values (int)	The number of the maximum lines per page (a default is 20)
"PAGERCOUNT": "3"	PAGERCOUNT		Arbitrary numerical values (int)	The number of the pager maximum displays (the number of displays before and behind the page concerned) (a
],				
"COND": [	COND		Array	Retrieval condition array
{			Object	Retrieval condition object
"COL": "shop.type",	COL	△	Table name . column name	It specifies the table name . column name given as conditions (a delimiter is a dot).
"OPE": "eq",	OPE	△	isnull	is null
			isnotnull	is not null
			eq	=
			nq	<>
			gt	>
			ge	>=
			lt	<
			le	<=
			%like	like '%xxxx'(Prefix search)
			%like%	like '%xxx%'(Middle coincidence search)
			like%	like 'xxxx%'(Suffix search)
			%notlike	not like '%xxx'
			%notlike%	not like '%xx%'
			notlike%	not like 'xxx%'
			in	in (1,2,3,4)
			notin	not in (1,2,3,4)
"VALUE": "_REQ.type"	VALUE	△	_REQ."Request parameter name" or _SES.LOGIN_USER."The column name of an authorization table " or _CONST."Constant"Key	It specifies the setting origin of a search condition. When specifying a request parameter, :"_REQ." parameter name. When authentication information is specified: The column name of a _SES.LOGIN_USER. authorization table When specifying a constant, :_CONST."Constant"Key(all the categories can be specified). In [category] not specifying, only key is specified.
}],				
"SORT": [	SORT		Array	Sorting condition array ※In the case of "select" or "selectAll", "METHOD" is effective
"shop.id desc",	–		Table name . Column name （asc/desc) or _REQ."Request parameter name" or _CONST."Constant"Key	It specifies the column name and ascending order or descending order for sorting (a default is an ascending order). When specifying a request parameter, :"_REQ." parameter name. When specifying a constant, :_CONST."Constant"Key(all the categories can be specified). In [category] not specifying, only key is specified.
...				
],				
"NOT_FOUND_ERR": "true"	NOT_FOUND_ERR		true/false	When data is not able to be extracted, it is specified whether it changes to 404 pages. (A default is false.) It does not become 404 errors.
}				

insert function

JSON(Object)	Parameter Name	Rquired	Setting Value	Description
{				
"TYPE": "crud",	TYPE		crud	Default
"METHOD": "insert",	METHOD	○	insert	It specifies it as "inset", when registering data.
"TARGET": "shop",	TARGET	○	Table name	It specifies the table name for processing.
"RESULT": "result1",	RESULT	○	Arbitrary character strings	Identifier specified by using data-bind-in Since data-bind-in cannot specify it when insert is specified by data-load, the following value is not referred to. * Make it into the name which becomes unique within a page.
"COL": [	COL		Array	Registration value array
{			Object	Registration object
"COL": "shop.type",	COL	△	Table name . column name	It specifies a table name . column name (a delimiter is a dot).
"VALUE": "_REQ.type"	VALUE	△	_REQ."Request parameter name" or _SES.LOGIN_USER."The column name of an authorization table " or _CONST."Constant"Key	It specifies a origin of setting. When specifying a request parameter, :"_REQ." parameter name. When authentication information is specified: The column name of a _SES.LOGIN_USER. authorization table When specifying a constant, :_CONST."Constant"Key(all the categories can be specified). In [category] not specifying, only key is specified.
}],				
"TOKEN": "true"	TOKEN		true/false	It is decided whether to check TOKEN (Default is true).
}				

update function

JSON（Object）	Parameter Name	Rquired	Setting Value	Description
[				
"TYPE": "crud",	TYPE		crud	default
"METHOD": "update",	METHOD	○	update	When updating data, it is specified as "update."
"TARGET": "shop",	TARGET	○	table name	It specifies the table name for processing.
"RESULT": "result1",	RESULT	○	Arbitrary character strings	Identifier specified by using data-bind-in and data-cond Since data-bind-in and data-cond cannot specify it when insert is specified by data-load, the following value is not referred to. ＊ Make it into the name which becomes unique within a page.
"COL": [	COL		Array	Update value array
			Object	Update object
{ "COL": "shop.type",	COL	△	table name . column name	It specifies a table name . column name (a delimiter is a dot).
"VALUE": "_REQ.type"	VALUE	△	_REQ."Request parameter name" or _SES.LOGIN_USER."The column name of an authorization table " or _CONST."Constant"Key	It specifies a origin of setting. When specifying a request parameter, :"_REQ." parameter name. When authentication information is specified: The column name of a _SES.LOGIN_USER. authorization table When specifying a constant, :_CONST."Constant"Key(all the categories can be specified). In [category] not specifying, only key is specified.
}],				
"COND": [	COND		Array	Update condition array
			Object	Update condition object
{ "COL": "shop.type",	COL	△	table name . column name	It specifies a table name . column name (a delimiter is a dot).
"OPE": "eq",	OPE	△	isnull	is null
			isnotnull	is not null
			eq	=
			nq	<>
			gt	>
			ge	>=
			lt	<
			le	<=
			%like	like '%xxxx'(Prefix search)
			%like%	like '%xxx%'(Middle coincidence search)
			like%	like 'xxxx%'(Suffix search)
			%notlike	not like '%xxx'
			%notlike%	not like '%xx%'
			notlike%	not like 'xxx%'
			in	in (1,2,3,4)
			notin	not in (1,2,3,4)
"VALUE": "_REQ.type"	VALUE	△	_REQ."Request parameter name" or _SES.LOGIN_USER."The column name of an authorization table " or _CONST."Constant"Key	It specifies a origin of setting. When specifying a request parameter, :"_REQ." parameter name. When authentication information is specified: The column name of a _SES.LOGIN_USER. authorization table When specifying a constant, :_CONST."Constant"Key(all the categories can be specified). In [category] not specifying, only key is specified.
}],				
"TOKEN": "true"	TOKEN		true/false	It is decided whether to check TOKEN (Default is true).
}				

delete function

JSON（Object）	Parameter Name	Rquired	Setting Value	Description
[				
"TYPE": "crud",	TYPE		crud	default
"METHOD": "delete",	METHOD	○	delete	When updating data, it is specified as "delete."
"TARGET": "shop",	TARGET	○	table name	It specifies the table name for processing.
"RESULT": "result1",	RESULT	○	Arbitrary character strings	Identifier specified by using data-cond Since data-cond cannot specify it when insert is specified by data-load, the following value is not referred to. ＊ Make it into the name which becomes unique within a page.
"COND": [	COND		Array	Deletion condition array
			Object	Deletion condition object
{ "COL": "shop.type",	COL	△	table name . column name	It specifies a table name . column name (a delimiter is a dot).
"OPE": "eq",	OPE	△	isnull	is null
			isnotnull	is not null
			eq	=
			nq	<>
			gt	>
			ge	>=
			lt	<
			le	<=
			%like	like '%xxxx'(Prefix search)
			%like%	like '%xxxx%'(Middle coincidence search)
			like%	like 'xxxx%'(Suffix search)
			%notlike	not like '%xxx'
			%notlike%	not like '%xx%'
			notlike%	not like 'xxx%'
			in	in (1,2,3,4)
			notin	not in (1,2,3,4)
"VALUE": "_REQ.type"	VALUE	△	_REQ."Request parameter name" or _SES.LOGIN_USER."The column name of an authorization table " or _CONST."Constant"Key	It specifies a origin of setting. When specifying a request parameter, :"_REQ." parameter name. When authentication information is specified: The column name of a _SES.LOGIN_USER. authorization table When specifying a constant, :_CONST."Constant"Key(all the categories can be specified). In [category] not specifying, only key is specified.
}],				
"TOKEN": "true"	TOKEN		true/false	It is decided whether to check TOKEN (Default is true).
}				

JSON(Object)	Parameter Name	Rquired	Setting Value	Description
[				
"TYPE": "procedure",	TYPE		procedure	It is specified as the case "procedure" where a procedure is used.
"TARGET": "shop",	TARGET	○	Stored procedure name	It specifies the stored procedure name for execution.
"RESULT": "result1",	RESULT	○	Arbitrary character strings	It is an identifier used by access to search results.
				※It is assumed a unique name in the page.
"OFFSET": "10",	OFFSET		Arbitrary numerical values (int)	It specifies the position which starts acquisition.
"LIMIT": "50",	LIMIT		Arbitrary numerical values (int)	It specifies the acquisition number.
"PAGER": {	PAGER		Object	Pager information object
				It specifies, when displaying a pager.
				* How to use pager information is referring to the [[supplementary] pager specification].
"PARAM_NAME": "page",	PARAM_NAME		Arbitrary character strings	It specifies the parameter name of page number.
				"page" is adopted when not specifying it.
"PERPAGE": "10",	PERPAGE		Arbitrary numerical values (int)	The number of the maximum lines per page (a default is 20)
"PAGERCOUNT": "3"	PAGERCOUNT		Arbitrary numerical values (int)	The number of the pager maximum displays (the number of displays before and behind the page concerned) (a
},				
"ARGS": [	ARGS		Array	Stored procedure argument array
{			Object	Stored procedure argument object
"NAME": "type",	COL	△	Arbitrary character strings	Name of stored procedure argument
				※It must be the same as the name at the time of the declaration of the stored procedure
"VALUE": "_REQ.type"	VALUE	△	_REQ."Request parameter name"	It specifies the setting origin of a stored procedure argument.
			or	When specifying a request parameter, :"_REQ." parameter name.
			_SES.LOGIN_USER."The column name of an	When authentication information is specified: The column name of a _SES.LOGIN_USER. authorization table
			authorization table "	When specifying a constant, :_CONST."Constant"Key(all the categories can be specified.).
			or	In [category] not specifying, only key is specified.
			_CONST."Constant"Key	
}],				
"NOT_FOUND_ERR": "true"	NOT_FOUND_ERR		true/false	When data is not able to be extracted, it is specified whether it changes to 404 pages.
				※When the return of stored procedure is not void, it is effective.
}				(Default does not become false.404 error.)

JSON(Object)	Parameter Name	Rquired	Setting Value	Description
[				
{				
"METHOD": "select",				
"TARGET": "company",				
"RESULT": "companyList"				
},			Refer to 「JSON(Object)」	
{				
"METHOD": "select",				
"TARGET": "shop",				
"RESULT": "shopList"				
}				
...				
]				

Sample

◆All the data is extracted and displayed from a company table at the time of a display. □
　　<head data-load='{"METHOD":"select","TARGET":"shop","RESULT":"result1"}'>

　　　
　　　<li data-iterator="result1"><b data-bind-out='[{"RESULT":"result1","TARGET":"text", "ARG": "shop.name"}]'>Sample
　　　

◆The data of the table to which the shop table and FK are set is extracted specifying the condition from the request parameter and the constant respectively when the screen is displayed.
　　<head data-load='{"METHOD":"selectAll","TARGET":"shop","RESULT":"result1","COND":[{"COL":"shop.type","OPE":"eq","VALUE":"_REQ.type"},{"COL":"shop.del_flag","OPE":"ne","VALUE":"_CONST.del"}]}'>
　　　<li data-iterator="result1" data-bind-out='{"RESULT":"result1","TARGET":"text","ARG":"shop.name"}'>shop name
　　　　
　　　　<li data-iterator="result1.shop.review" data-bind-out='{"RESULT":"result1","TARGET":"text","ARG":"review.content"}'>review
　　　　
　　　

data-func

search function

JSON（Object）	Parameter Name	Rquired	Setting Value	Description
{				
"TYPE": "crud",	TYPE		crud	It is specified as the case "crud" where a CRUD function is used. (Default)
"METHOD": "select",	METHOD	○	select	It specifies, when extracting data from the table which specified arbitrary search conditions and was set as target.
			selectAll	It specifies, when specifying arbitrary search conditions and extracting all the data including a related table (FK).
			selectByPk	One record of a target table is extracted. data-bind-out can be described, without specifying data-iterator. When this method is specified, it becomes an error when the extraction result is not one.
"TARGET": "shop",	TARGET	○	Table name	It specifies the table name for processing.
"RESULT": "result1",	RESULT	○	Arbitrary character strings	It is an identifier used by access to search results. ※It is assumed a unique name in the page.
"OFFSET": "10",	OFFSET		Arbitrary numerical values (int)	It specifies the position which starts acquisition ※In the case of "select" or "selectAll", "METHOD" is effective ※The line returned is not guaranteed when not specifying "SORT".
"LIMIT": "50",	LIMIT		Arbitrary numerical values (int)	It specifies the acquisition number. ※In the case of "select" or "selectAll", "METHOD" is effective ※The line returned is not guaranteed when not specifying "SORT".
"PAGER": {	PAGER		Object	Pager information object It is specified when displaying a pager. ※The directions for pager information are referring to the [[supplementary] pager specification]. ※In the case of "select" or "selectAll", "METHOD" is effective
"PARAM_NAME": "page",	PARAM_NAME		Arbitrary character strings	It specifies the parameter name of page number. "page" is used when not specifying.
"PERPAGE": "10",	PERPAGE		Arbitrary numerical values (int)	The number of the maximum lines per page (a default is 20)
"PAGERCOUNT": "3"	PAGERCOUNT		Arbitrary numerical values (int)	The number of the pager maximum displays (the number of displays before and behind the page concerned) (a
},				
"COND": [	COND		Array	Retrieval condition array
{			Object	Retrieval condition object
"COL": "shop.type",	COL	△	Table name . column name	It specifies the table name . column name given as conditions (a delimiter is a dot).
"OPE": "eq",	OPE	△	isnull	is null
			isnotnull	is not null
			eq	=
			nq	<>
			gt	>
			ge	>=
			lt	<
			le	<=
			%like	like '%xxxx'(Prefix search)
			%like%	like '%xxx%'(Middle coincidence search)
			like%	like 'xxxx%'(Suffix search)
			%notlike	not like '%xxx'
			%notlike%	not like '%xx%'
			notlike%	not like 'xxx%'
			in	in (1,2,3,4)
			notin	not in (1,2,3,4)
"VALUE": "_REQ.type"	VALUE	△	_REQ."Request parameter name" or _SES.LOGIN_USER."The column name of an authorization table " or _CONST."Constant"Key	It specifies the setting origin of a search condition. When specifying a request parameter, :"_REQ." parameter name. When authentication information is specified: The column name of a _SES.LOGIN_USER. authorization table When specifying a constant, :_CONST."Constant"Key(all the categories can be specified.). In [category] not specifying, only key is specified.
}],				
"SORT": [	SORT		Array	Sorting condition array ※In the case of "select" or "selectAll", "METHOD" is effective
"shop.id desc",	–		Table name . Column name （asc/desc) or _REQ."Request parameter name" or _CONST."Constant"Key	It specifies the column name and ascending order or descending order for sorting (a default is an ascending order). When specifying a request parameter, :"_REQ." parameter name. When specifying a constant, :_CONST."Constant"Key(all the categories can be specified.). In [category] not specifying, only key is specified.
…				
],				
"NOT_FOUND_ERR": "true"	NOT_FOUND_ERR		true/false	When data is not able to be extracted, it is specified whether it changes to 404 pages. (A default is false.) It does not become 404 errors.
}				

insert function

JSON（Object）	Parameter Name	Rquired	Setting Value	Description
{				
"TYPE": "crud",	TYPE		crud	Default
"METHOD": "insert",	METHOD	○	insert	It specifies it as "inset", when registering data.
"TARGET": "shop",	TARGET	○	Table name	It specifies the table name for processing.
"RESULT": "result1",	RESULT	○	Arbitrary character strings	Identifier specified by using data-bind-in Since data-bind-in cannot specify it when insert is specified by data-load, the following value is not referred to. * Make it into the name which becomes unique within a page.
"COL": [	COL		Array	Registration value array
{			Object	Registration object
"COL": "shop.type",	COL	△	Table name . column name	It specifies a table name . column name (a delimiter is a dot).
"VALUE": "_REQ.type"	VALUE	△	_REQ."Request parameter name" or _SES.LOGIN_USER."The column name of an authorization table " or _CONST."Constant"Key	It specifies a origin of setting. When specifying a request parameter, :"_REQ." parameter name. When authentication information is specified: The column name of a _SES.LOGIN_USER. authorization table When specifying a constant, :_CONST."Constant"Key(all the categories can be specified.). In [category] not specifying, only key is specified.
}],				
"TOKEN": "true"	TOKEN		true/false	It is decided whether to check TOKEN (Default is true).
}				

update function

JSON（Object）	Parameter Name	Rquired	Setting Value	Description
{				
"TYPE": "crud",	TYPE		crud	default
"METHOD": "update",	METHOD	O	update	When updating data, it is specified as "update."
"TARGET": "shop",	TARGET	O	table name	It specifies the table name for processing.
"RESULT": "result1",	RESULT	O	Arbitrary character strings	Identifier specified by using data-bind-in and data-cond Since data-bind-in and data-cond cannot specify it when insert is specified by data-load, the following value is not referred to. * Make it into the name which becomes unique within a page.
"COL": [	COL		Array	Update value array
{			Object	Update object
"COL": "shop.type",	COL	△	table name . column name	It specifies a table name . column name (a delimiter is a dot).
"VALUE": "_REQ.type"	VALUE	△	_REQ."Request parameter name" or _SES.LOGIN_USER."The column name of an authorization table" or CONST."Constant"Key	It specifies a origin of setting. When specifying a request parameter, :"_REQ." parameter name. When authentication information is specified: The column name of a _SES.LOGIN_USER. authorization table When specifying a constant, :_CONST."Constant"Key(all the categories can be specified.). In [category] not specifying, only key is specified.
}],				
"COND": [	COND		Array	Update condition array
{			Object	Update condition object
"COL": "shop.type",	COL	△	table name . column name	It specifies a table name . column name (a delimiter is a dot).
"OPE": "eq",	OPE	△	isnull	is null
			isnotnull	is not null
			eq	=
			nq	<>
			gt	>
			ge	>=
			lt	<
			le	<=
			%like	like '%xxxx'(Prefix search)
			%like%	like '%xxx%'(Middle coincidence search)
			like%	like 'xxxx%'(Suffix search)
			%notlike	not like '%xxx'
			%notlike%	not like '%xx%'
			notlike%	not like 'xxx%'
			in	in (1,2,3,4)
			notin	not in (1,2,3,4)
"VALUE": "_REQ.type"	VALUE	△	_REQ."Request parameter name" or _SES.LOGIN_USER."The column name of an authorization table" or CONST."Constant"Key	It specifies a origin of setting. When specifying a request parameter, :"_REQ." parameter name. When authentication information is specified: The column name of a _SES.LOGIN_USER. authorization table When specifying a constant, :_CONST."Constant"Key(all the categories can be specified.). In [category] not specifying, only key is specified.
}],				
"TOKEN": "true"	TOKEN		true/false	It is decided whether to check TOKEN (Default is true).
}				

delete function

JSON（Object）	Parameter Name	Rquired	Setting Value	Description
{				
"TYPE": "crud",	TYPE		crud	default
"METHOD": "delete",	METHOD	O	delete	When updating data, it is specified as "delete."
"TARGET": "shop",	TARGET	O	table name	It specifies the table name for processing.
"RESULT": "result1",	RESULT	O	Arbitrary character strings	Identifier specified by using data-cond Since data-cond cannot specify it when insert is specified by data-load, the following value is not referred to. * Make it into the name which becomes unique within a page.
"COND": [	COND		Array	Deletion condition array
{			Object	Deletion condition object
"COL": "shop.type",	COL	△	table name . column name	It specifies a table name . column name (a delimiter is a dot).
"OPE": "eq",	OPE	△	isnull	is null
			isnotnull	is not null
			eq	=
			nq	<>
			gt	>
			ge	>=
			lt	<
			le	<=
			%like	like '%xxxx'(Prefix search)
			%like%	like '%xxx%'(Middle coincidence search)
			like%	like 'xxxx%'(Suffix search)
			%notlike	not like '%xxx'
			%notlike%	not like '%xx%'
			notlike%	not like 'xxx%'
			in	in (1,2,3,4)
			notin	not in (1,2,3,4)
"VALUE": "_REQ.type"	VALUE	△	_REQ."Request parameter name" or _SES.LOGIN_USER."The column name of an authorization table" or CONST."Constant"Key	It specifies a origin of setting. When specifying a request parameter, :"_REQ." parameter name. When authentication information is specified: The column name of a _SES.LOGIN_USER. authorization table When specifying a constant, :_CONST."Constant"Key(all the categories can be specified.). In [category] not specifying, only key is specified.
}],				
"TOKEN": "true"	TOKEN		true/false	It is decided whether to check TOKEN (Default is true).
}				

stored procedure execution

JSON（Object）	Parameter Name	Rquired	Setting Value	Description
{				
"TYPE": "procedure",	TYPE		procedure	It is specified as the case "procedure" where a procedure is used.
"TARGET": "shop",	TARGET	○	Stored procedure name	It specifies the stored procedure name for execution.
"RESULT": "result1",	RESULT	○	Arbitrary character strings	It is an identifier used by access to search results. ※It is assumed a unique name in the page.
"OFFSET": "10",	OFFSET		Arbitrary numerical values (int)	It specifies the position which starts acquisition.
"LIMIT": "50",	LIMIT		Arbitrary numerical values (int)	It specifies the acquisition number.
"PAGER": [	PAGER		Object	Pager information object It specifies, when displaying a pager. * How to use pager information is referring to the [[supplementary] pager specification].
"PARAM_NAME": "page",	PARAM_NAME		Arbitrary character strings	It specifies the parameter name of page number. "page" is adopted when not specifying it.
"PERPAGE": "10",	PERPAGE		Arbitrary numerical values (int)	The number of the maximum lines per page (a default is 20)
"PAGERCOUNT": "3"	PAGERCOUNT		Arbitrary numerical values (int)	The number of the pager maximum displays (the number of displays before and behind the page concerned) (a
],				
"ARGS": [	ARGS		Array	Stored procedure argument array
{			Object	Stored procedure argument object
"NAME": "type",	COL	△	Arbitrary character strings	Name of stored procedure argument *It must be the same as the name at the time of the declaration of the stored procedure
"VALUE": "_REQ.type"	VALUE	△	_REQ."Request parameter name" or _SES.LOGIN_USER."The column name of an authorization table" or _CONST."Constant"Key	It specifies the setting origin of a stored procedure argument. When specifying a request parameter, :"_REQ." parameter name. When authentication information is specified: The column name of a _SES.LOGIN_USER. authorization table When specifying a constant, :_CONST."Constant"Key(all the categories can be specified.). In [category] not specifying, only key is specified.
}],				
}				

メール送信機能

JSON（Object）	Parameter Name	Rquired	Setting Value	Description
{				
"TYPE": "sendmail",	TYPE	○	sendmail	It specifies it as the case "sendmail" where e-mail transmission is performed.
"METHOD": "inquery_mail",	METHOD	○	E-mail template name	It is a file name (*. ftl) of the e-mail template stored in the templates folder.
"TARGET": "inquery",	TARGET		table name	It specifies the table name which saves transmitting contents.It is unnecessary when not saving transmitting
"RESULT": "result1",	RESULT	○	Arbitrary character strings	It is an identifier used when the content of the transmission is specified. ※It is assumed a unique name in the page.
"ADDRESS_COL": "shop.address",	ADDR_COL	△	Addressee address column	It specifies the table and column name of an address of an addressee.
"ID_COL": "ids",	ID_COL	△	Sender id preservation column	It specifies the column name which saves PK of the table in which the address of the sender was saved.
"TO": [	TO	△	key array of the to address	Indispensable, when not specifying id_col It specifies Category.Key of a system_constan table, when acquiring to mail address from a system constant and setting it up. When specifying a constant, :_CONST."Constant"Key(all the categories can be specified.). In [category] not specifying, only key is specified.
"admin_to",				
...				
],				
"CC": [	CC		key array of the Cc address	It specifies Category.Key of a system_constan table, when acquiring cc mail address from a system constant and setting it up. It specifies Category.Key of a system_constan table, when acquiring cc mail address from a system constant and setting it up. In [category] not specifying, only key is specified.
"admin_cc",				
...				
],				
"BCC": [	BCC	–	key array of the Bcc address	It specifies Category.Key of a system_constan table, when acquiring a bcc mail address from a system constant and When specifying a constant, : CONST."Constant"Key(all the categories can be specified.). In [category] not specifying, only key is specified.
"admin_bcc",				
...				
]				
}				

JSON（Array）	Parameter Name	Rquired	Setting Value	Description

[
  {
    "TYPE": "crud",
    "METHOD": "select",
    "TARGET": "shop",
    "RESULT": "result1"
  },
  {
    "TYPE": "sendmail",
    "METHOD": "inquery_mail",
    "RESULT": "result2",
    "TO": [
      "admin_to"
    ]
  }
  ...
]
			Refer to「JSON（Object）」	

Sample

data-cond

*select a value from setting value designated.

JSON(Object)	Parameter Name	Rquired	Setting Value	Description
{				
"RESULT": "result1",	RESULT	○	Result identifier specified with data-func	It specifies in order to identify the conditions over which data-func it is.
"COL": "shop.type",	COL	○	Table name . column name	It specifies the table name . column name given as conditions (a delimiter is a dot).
"OPE": "eq",	OPE	○	isnull	is null
			isnotnull	is not null
			eq	=
			nq	<>
			gt	>
			ge	>=
			lt	<
			le	<=
			%like	like '%xxxx'(Prefix search)
			%like%	like '%xxx%'(Middle coincidence search)
			like%	like 'xxxx%'(Suffix search)
			%notlike	not like '%xxx'
			%notlike%	not like '%xx%'
			notlike%	not like 'xxx%'
			in	in (1,2,3,4)
			notin	not in (1,2,3,4)
"VALUE": "_CONST.type"	VALUE		Table name . Column name (asc/desc) or _REQ."Request parameter name" or _CONST."Constant"Key	It is specified when setting an initial value to a search condition. It is not specified when always carrying out on condition of the input of an input tag. When specifying a request parameter, :"_REQ." parameter name. When authentication information is specified: The column name of a _SES.LOGIN_USER. authorization table When specifying a constant, :_CONST."Constant"Key(all the categories can be specified.). In [category] not specifying, only key is specified.
}				

Sample
◆When an input value is made into a search condition
<form action="../../html/shop/list.html" method="POST" id="listForm" data-func='{"TYPE":"crud","TARGET":"shop","METHOD":"select","RESULT":"result1"}'>
<input type="text" data-cond='{"RESULT":"result1","COL":"shop.name","OPE":"%like%"}'>
◆When a request parameter is made into an early search condition
<input type="hidden" data-cond='{"RESULT":"result1","COL":"shop.id","OPE":"in","VALUE":"_REQ.ids"}' />
◆en authentication information is made into an early search condition □
<input type="hidden" data-cond='{"RESULT":"result1","COL":"shop.id","OPE":"eq","VALUE":"_SES.LOGIN_USER.id"}' />
◆When CONST is made into an early search condition
<input type="hidden" data-cond='{"RESULT":"result1","COL":"shop.flag","OPE":"eq","VALUE":"_CONST.flag"}' />

data-iterator

Parameter Name	Rquired	Setting Value	Description
Result identifier(.FK table name)	○	Result identifier or Result identifier.PAGE_NO_LIST	Result identifier specified with data-load or data-func When the child table in which FK was specified for METHOD to target by "selectAll" exists, a multiplex repetition is possible by giving the table name of a child table. When specifying the paging number list of pager information, "result identifier .PAGE_NO_LIST" is specified.

Sample

◆The example of multiplex iterator
    <ul class="TMP_L_0001" data-case='{"TYPE":"size","ARG":["result1","gt","0"]}'>
     <li data-iterator="result1" data-bind='{"TARGET":"text","ARG":"result1.shop.name"}'>shop name

       <li data-iterator="result1.shop.review" data-bind='{"TARGET":"text","ARG":"result1.shop.review.content"}'>review

特筆事項
・Brother elements other than the element in which data-iterator was specified are deleted (except for the case where data-keep is set up). □

data-keep

Parameter Name	Rquired	Setting Value	Description
"true" fixation			

Sample

<ul class="TMP_L_0001" data-case='{"TYPE":"size","ARG":["result1","gt","0"]}'>
  <li data-keep="true">shop name
  <li data-iterator="result1" data-bind='{"TARGET":"text","ARG":"result1.shop.name"}'>shop name

data-bind-in

JSON（Object）	Parameter Name	Rquired	Setting Value	Description
"RESULT": "result1", "ARG": "shop.type", "VALUE": "_CONST.type"	RESULT	○	Result identifier specified with data-func	It specifies in order to identify the conditions over which data-func it is.
	ARG	○	Table name . column name	It specifies the table name . column name given as conditions (a delimiter is a dot).
	VALUE		Table name . Column name (asc/desc) or _REQ."Request parameter name" or _CONST."Constant"Key	It is specified when setting an initial value to a search condition. It is not specified when always carrying out on condition of the input of an input tag. When specifying a request parameter, :"_REQ." parameter name. When authentication information is specified: The column name of a _SES.LOGIN_USER. authorization table When specifying a constant, :_CONST."Constant"Key(all the categories can be specified.). In [category] not specifying, only key is specified.
}				

Sample
◆input text
<form action="../../html/shop/list.html" method="POST" id="listForm" data-func='{"TYPE":"crud","TARGET":"shop","METHOD":"insert","RESULT":"result1"}'>
<input type="text" data-bind-in='{"RESULT":"result1","ARG": "shop.name"}'>shop name
<input type="hidden" data-bind-in='{"RESULT":"result1","ARG": "shop.type","DEFAULT":"_CONST.type"}'>
◆textarea
<form action="../../html/shop/list.html" method="POST" id="listForm" data-func='{"TYPE":"crud","TARGET":"shop","METHOD":"update","RESULT":"result1"}'>
<textarea rows="4" cols="40" data-bind-in='{"RESULT":"result1","ARG": "shop.summary"}'>Outline of shop</textarea>

data-bind-out

*select a value from setting value designated.

JSON(Object)	Parameter Name	Rquired	Setting Value	Description
{				
"RESULT": "result1"	RESULT		result identifier designated at data-load or data-func	Designate to distingish which result of data-load and data-func should be output. If the contents that will be binded are NOT a result of data-load or data-func, don't designate.
"TARGET": "text",	TARGET	○	attribute name for bind or"text" or"inner"	Designate the attribute name subjected to bind as src, href and so on. In the case of designating "text", text of factor setted data-bind attribute will be replaced. In the case of designating "inner", the attribute under it will be replaced.
"FORMAT": "/{0}/{1}/",	FORMAT		arbitrary character string or DATE[format] or NUMBER[format]	In the case of unspecified a value, bind the data extracted as it is, but if there is replaced string{0} in target, set target format. In the other case that target contains [_ORIGINAL], give the original attribute data of target. Details about date and numerical value format will be mentioned later.
"ARG": "shop.name",	ARG	△	table name.column name or _REQ.request parameter name or _SES.LOGIN_USER.Attestation table	In the case that column of taeget is only one: don't designate "args" but "arg". In the case of binding the values of DB: designate the table name and column name of where you get the values. In the case of binding request parameter: designate _REQ. parameter name. In the case of designating an attestarion information: set the column name _SES.LOGIN_USER.attestation table. In the case of binding constants: "_CONST."constantKey. (only Key of view category can be.) In the case of designating a pager information: "_PAGER".pager information Key(refer to [【supplement】
"ARGS": ["shop.id",	ARGS	△	–	designate parameter array which you bind.(It is necessary to disignate the number of replaced strings in format)
			table name.column name or _REQ.request parameter name or _SES.LOGIN_USER.Attestation table	In the case of binding DB values: the table name and column name of where you get the values. In the case of binding request parameter: designate _REQ. parameter name. In the case of designating an attestarion information: set the column name _SES.LOGIN_USER.attestation table. In the case of binding constants: "_CONST."constantKey. (only Key of view category can be.) In the case of designating a pager information: "_PAGER".pager information Key(refer to [【supplement】 pager]sheet)
"VALUE": "_CONST.type"	VALUE		_REQ.request parameter name or _CONST.constant Key	designate this when you set a default. ※you can designate only the time when you designate ARG. In the case of designating request parameter : "_REQ."parameter name In the case of binding constants: "_CONST."constantKey. (only Key of view category can be.)
...				
],				
"MAP_KEY": "_CONST.view.stars"	MAP_KEY		_CONST.constant Key	set value based on contents designated in ARG as Key and get an information defined on _webappos_system_constant.
"ESCAPE":"false",	ESCAPE		true	If HTML tag was registered with DB, escape the HTML tag. (default)
			false	If HTML tag was registered with DB, do NOT escape the HTML tag. (default) ※you can use when you bind the values of DB.
"ERASE_TAG":"false".	ESCAPE		true	In the case of binding strings that contain HTML tag, delete HTML tag.
			false	In the case of binding strings that contain HTML tag, delete HTML tag.
}				

JSON(Array)	Parameter Name	require	Setting Value	Description
[{ "RESULT": "result1", "TARGET": "href", "FORMAT": "/{0}/{1}", "ARGS": ["id", "category_code"] }, { "RESULT": "result1", "TARGET": "text", "ARG": "shop.name" }]				Refer to 「JSON(Object)」

Sample

◆text(overwrite)
```
<sapn data-bind-out='{"RESULT":"result1","TARGET":"text", "ARG": "shop.name"}'>shop_name</span>
```

◆text(replace partly)
```
<sapn data-bind-out='{"RESULT":"result1","TARGET":"text", "ARG": "shop.name"}'>{0} recommended information </span>
```

◆text (a constant)
```
<sapn data-bind-out='{"TARGET":"text", "ARG": "_CONST.titile"}'>NEWS</span>
```

◆text(a date format)
```
<sapn data-bind-out='{"RESULT":"result1","TARGET":"text", "FORMAT":"DATE[yyyy/MM/dd]","ARG": "shop.publish_date"}'>2013/12/03</span>
```

◆text(numerical value format, comma)
```
<sapn data-bind-out='{"RESULT":"result1","TARGET":"text", "FORMAT":"NUMBER[#,##0]","ARG": "shop.employees"}'>123,456</span>
```

◆a[href]
```
<a href="../../html/shop/list.html" data-bind-out='[{"RESULT":"result1","TARGET":"href","FORMAT":"[_ORIGINAL]type/{0}/","ARG":"type.name"}]'>Android</a>
```

◆img[src]
```
<img src="/mok/menu.png" data-bind-out='[{"RESULT":"result1","TARGET":"src","FORMAT":"/images/{0}/{1}.png","ARGS":["shop.id","type.id"]}]' alt="menu" />
```

Date and numerical value format

Date and numerical format can be used in case that ARG is designated.(If you designate "FORMAT"+"ARGS", you can not designate a date and numerikal value format at "FORMAT".)

◆Date format pattern
DATE[format]
describe the pattern that mentioned in [【supplement】format]sheet.
ex.)

yyyy/M/d	→ 2011/8/3

◆Numerical value format pattern
NUMBER[format]
describe the pattern that mentioned in [【supplement】format]sheet.
ex.)

#,###.##	→ 123,456,789.57
00,000.00	→ 01,234.50

data-case

JSON（Object）	Parameter Name	equire	Setting Value	Dsscription
{				
"TYPE": "compare",	TYPE	O	compare	compare with arbitrary value
			size	compare the size of retuen list
				use this with data-iterator
"OPE": "or",	OPE		and	designate conditional operator in the case which data-case is designated in array
			or	default is "and"
"ARG": [	ARG	O	–	array that designate a comparison condition

"companyList.company.name",	–	O	a target compared with	In the case of "compare":
				  In the case of comparing DB values: the table name and column name of where you get the values
				  In the case of comparing request parameter: "_REQ." parameter name
				  In the case of comparing constants: "_CONST."constantKey (only Key of view category can be.)
				  In the case of comparing session values : "_SES."Key
				  In the case of comparing pager values: result identifier."_PAGER".pager information Key
				
				In the case of "size":
				  result identifier of data-func or data-load

			result identifier._VALUE	In the case of comparing with List value: "result identifier._VALUE"
			result identifier._STATUS.index	In the case of comparing with List status (count from 0)
			result identifier._STATUS.count	In the case of comparing with List status (count from 1)
			result identifier._STATUS.odd	In the case of comparing with List status (if it is a line of odd number)
			result identifier._STATUS.even	In the case of comparing with List status (if it is a line of even number)
			result identifier._STATUS.first	In the case of comparing with List status (if it is a first line)
			result identifier._STATUS.last	In the case of comparing with List status (if it is a last line)
			result identifier._STATUS.modulu	In the case of comparing with List status (the rest of it devided by n)
"eq",	–	O	eq	=
			ne	<>
			gt	>
			ge	>=
			lt	<
			le	<=
"1"	–		arbitrary value	comparizon value
]				
}				

JSON（Array）	Parameter Name	equire	Setting Value	Description

[
 {
  "TYPE": "compare",
  "ARG": [
   "result1.shop.type",
   "eq",
   "1"
  ]
 },
 {
  "TYPE": "compare",
  "ARG": [
   "result1.shop.name",
   "ne",
   "null"
  ]
 }
 …
]
				Refer to「JSON（Object）」

Sample

data-erase

Parameter	Required	Setting Value	Description
set a cope deleated	〇	own	delete the factor designated and factor under that which data-erasure is designated
		child	delete the factor and the factor under that which data-erasure is designated

data-url

JSON(Array)	Parameter Name	Ruquire	Setting Value	Description
[				
"/shop/detail/id/{0}",	–	○	Request URL Pattern	designate a path after context set variable parameter {0} and Key of parameter the path designated just before this. In this example, parameter Key is id.
....				
]				

Sample

◆a SINGLE url pattern and NO parameter
    <head data-url='["/shop/search/list/"]'>
    URL：http://xxxxxxx.com/shop/search/list/

◆the PLURAL url pattern and NO parameter
    <head data-url='["/shop/search/list/","/shop/search/list/a/"]'>
    URL：http://xxxxxxxx.com/shop/search/list/　or　http://xxxxxxxx.com/shop/search/list/a/

◆a SINGLE url pattern and parameter
    <head data-url='["/shop/detail/id/{0}/"]'>
    URL：http://xxxxxxxx.com/shop/detail/id/47/

◆a SINGLE url pattern and the PLURAL parameter
    <head data-url='["/shop/detail/id/{0}/type/{0}/"]'>
    URL：http://xxxxxxxx.com/shop/detail/id/47/type/1/

◆the PLURAL url pattern and parameter
    <head data-url='["{_ORIGINAL}id/{0}/","{_ORIGINAL}year/{0}/"]'>
    URL：http://xxxxxxxx.com/shop/detail/id/47/ or URL：http://xxxxxxxx.com/shop/detail/year/2003/
    ※Apply the following rule of HTML file path to {_ORIGINAL} and set strings that added a parameter as URL.

◆URL that is designeted data-url be fixed by folder composition which html file is placed.
    /[folder name]-[subfolder neme]/[html file name]/
    ・¥shop¥list.html
    http://xxxxxxxx.com/shop/list/

    ・¥shop¥search¥list.html
    http://xxxxxxxx.com/shop-search/list/

    ・¥shop¥search¥type¥list.html
    http://xxxxxxxx.com/shop-search-type/list/

data-auth

Parameter	Required	Setting Value	Description
fix	"true"		

Sample
<head data-auth='true'>
</head>

data-image

JSON(Object)	Parameter Name	Require	Setting Value	Description
{				
"RESULT": "result1"	RESULT		result identifier of data-load or data-func	designate for decision which result of data-load and data-func should be outputted. DON'T designate this if contents binded is NOT a result of data-load and data-func.
"TARGET": "data-orignal",	TARGET		attribute name of target	if it have NOT designated yet, bind to src.
"ARG": "event.path",	ARG	○	table name.column name or _REQ.request parameter name or CONST.constant Key	Binding DB values：table name. column name where you got it Binding request parameter：" _REQ."parameter Binding constants：" CONST."constant Key(you can use only a Key of veiw category)
"W": "100",	W		100	designate in width by pixel for changing the size of an uploaded picture.
"H": "150",	H		150	designate in height by pixel for changing the size of an uploaded picture.
}				

Sample

pager specification

specification

◆PAGER specification

be returned pager information if PAGER object is designated at select method of data-load or data-func.

the way of reference：result identifier._PAGER

available information：

RECORD_COUNT：the number of whole records

PER_PAGE：maximum per a page(=PERPAGE)

RECORD_BEGIN_NO：record number of start of showing(OFFSET)

RECORD_END_NO：record number of end of showing

PAGE_NO：pressent page number

PREV_PAGE_NO：previous page number (If there is not previous page, return 0.)

NEXT_PAGE_NO：next page number (If there is not next page, return 0.)

PAGING_BEGIN_NO：paging number of start of showing

PAGING_END_NO：paging number of end of showing

MAX_PAGE_NO：the last page number

PAGE_NO_LIST：paging number list

◆sample

previous <u>1 2 3 4 5 6 7</u> next

・HTML

```
<nav>
 <ul id="pager">
  <li data-keep="true" data-case='{"TYPE":"compare","ARG":["result1._PAGER.PREV_PAGE_NO","ne","0"]}'>
   <a href="../../html/list.html" data-bind-out='{"RESULT":"result1","TARGET":"href","ARG":"_PAGER.PREV_PAGE_NO","FORMAT":"{_ORIGINAL}/page/{0}/"}'>前のページ</a>
  </li>
  <li data-iterator='result1.PAGE_NO_LIST'>
   <a href="../../html/list.html" data-bind-out='[{"TARGET":"text"},{"TARGET":"href","ARG":,"","FORMAT":"{_ORIGINAL}/page/{0}/"}]'>1</a>
  </li>
  <li data-keep="true" data-case='{"TYPE":"compare","ARG":["result1._PAGER.NEXT_PAGE_NO","ne","0"]}'>
   <a href="../../html/list.html" data-bind-out='{"RESULT":"result1","TARGET":"href","ARG":"_PAGER.NEXT_PAGE_NO","FORMAT":"{_ORIGINAL}/page/{0}/}'>次のページ</a>
  </li>
 </ul>
</nav>
```

◆Date format

word	Meaning	example
yy	Two-digit year	1996 → 96
yyyy	Four-digit year	1996
M	Month	8 → 8
MM	Month	8 → 08
D	Day for the year	1996-01-09 → 9
DDD	Day for the year	1996-01-09 → 009
d	One-digit day for days below 10	3 → 3
dd	Two-digit day	3 → 03
w	Week for the year	2011-8-30 → 36
W	Week for the month	2011-8-30 → 5
E	day of week	2011-8-30→ Tuesday
F	Day number for the month	2011/8/30 →5
a	am,pm	13:00 → pm
h	hour(12hour)	13:00→1
hh	hour(12hour)	13:00→01
H	hour(24hour)	03:00→3
HH	hour(24hour)	03:00→03
m	minute	00:03→3
mm	minute	00:03→03
s	second	00:00:03→3
ss	second	00:00:03→03
S	milliseconds	00:00:00.003→3
SSS	milliseconds	00:00:00.003→003

◆Number format

word	memo
0	This shows the numerical value of one digit. "0" is displayed when there is no numerical value in the digit.
#	This shows the numerical value of one digit. It becomes blank when there is no numerical value in the digit.
.	Decimal point
,	Comma Separated Value
-	minus
%	100 numerical values are multiplied and percent is

Attestation Information

Specification

・if it succeed form attestation or OAuth attestation, attestation information will be session
 	
■access to attestation Information
   _SES.LOGIN_USER.*column name of attestation table*
   attestation table： the table designated at auth.table.name of application.properties