OpenClovis
Software Development Kit (SDK)
Service Description and API Reference for
Rule Based Engine (RBE) Service

For OpenClovis SDK Release 2.3 V0.4

Document Revision Date: March 08, 2007

Contents

Chapter 1

Functional Overview

The OpenClovis Rule Based Engine (RBE) provides a mechanism to create rules to be applied to the system instance data, based on simple expressions.

An expression consists of a mask and a value. These expressions are evaluated on user data and a boolean value is generated for the decision process.

For instance, RBE is used by the Event Service to support filter-based subscriptions. The event is published with a pattern that is matched against the filter provided by the subscribers. Only those subscribers that match successfully are notified. The RBE library provides simple bit-based matching based on the flags specified.

1.1 Interaction with other components

The RBE engine forms a general purpose library that can be used by any other software component. The RBE library does not depend on any component.

Chapter 2

Service Model

TBD

Chapter 3

Service APIs

3.1 Type Definitions

3.1.1 ClRuleExprT

```
typedef struct RuleExpr{
            ClUint8T flags;
            ClUint8T len;
            ClUint16T offset;
            struct RuleExpr *next;
                  ClUint8T Byte [1]
                  ClUint32T Int [1]
            BI_u
} ClRuleExprT;
```

The structure, `ClRuleExprT`, contains the rule to filter data. It provides the expression definition. The attributes of the structure are:

- *flags* - Architecture and other flags.
- *len* - Expression length in multiples of four bytes.
- **next* - Multiple RBEs can be chained to build complex expression.
- *offset* - Offset with the data where expression is applied.
- *Byte*, *Int* - Expression mask and value array.

3.1.2 ClRuleExprFlagsT

```
typedef enum{
            CL_RULE_LITTLE_END = 0x1,
            CL_RULE_BIG_END = 0x2,
            CL_RULE_NON_ZERO_MATCH = 0x4,
            CL_RULE_MATCH_EXACT = 0x8,
            CL_RULE_EXPR_CHAIN_AND = 0x10,
} ClRuleExprFlagsT;
```

The values of the `ClRuleExprFlagsT` enumeration, contain the expression qualification. It filters the data based on the rule. The attributes of the enumeration are:

- *CL_RULE_LITTLE_END* - Little endian
- *CL_RULE_BIG_END* - Big endian.
- *CL_RULE_NON_ZERO_MATCH* - Non-zero match.
- *CL_RULE_MATCH_EXACT* - Exact match.
- *CL_RULE_EXPR_CHAIN_AND* - Appending two RBE expressions with AND relation.

3.2 Functional APIs

3.2.1 clRuleExprAllocate

clRuleExprAllocate

Synopsis:

Allocates RBE expression.

Header File:

clRuleApi.h

Syntax:

```
    ClRcT clRuleExprAllocate(
                CL_IN ClUint8T len,
                CL_OUT ClRuleExprT** ppExpr);
```

Parameters:

len: (in) Length of the mask or value in multiples of four bytes.

ppExpr: (out) Allocated expression is returned in this parameter.

Return values:

CL_OK: The function executed successfully.

CL_RULE_RC(CL_ERR_NULL_POINTER): `ppExpr` contains a NULL pointer.

CL_RULE_RC(CL_ERR_NO_MEMORY): An error has occurred.

Description:

This function allocates an RBE expression and initializes it appropriately.

Library File:

libClUtils

Related Function(s):

clRuleExprDeallocate

3.2.2 clRuleExprDeallocate

clRuleExprDeallocate

Synopsis:
Frees memory or structures used by an RBE expression.

Header File:
clRuleApi.h

Syntax:

```
    ClRcT clRuleExprDeallocate(
                CL_IN ClRuleExprT* pExpr);
```

Parameters:
Expr: (in) RBE expression to be freed.

Return Values:
CL_OK: The function executed successfully.

CL_RULE_RC(CL_ERR_NULL_POINTER): `pExpr` contains a NULL pointer.

CL_RULE_RC(CL_ERR_NO_MEMORY): An error has occurred.

Description:
This function frees the memory or structures used by RBE expression.

Library File:
libClUtils

Related Function(s):
clRuleExprAllocate

3.2.3 clRuleExprAppend

clRuleExprAppend

Synopsis:

Appends a RBE expression.

Header File:

clRuleApi.h

Syntax:

```
ClRcT clRuleExprAppend (
            CL_IN ClRuleExprT* pFirstExpr,
            CL_IN ClRuleExprT* pNextExpr);
```

Parameters:

pFirstExpr: (in) RBE expression on which `pNextExpr` needs to be appended.

pNextExpr: (in) RBE expression that is appended to `pFirstExpr`.

Return values:

CL_OK: The function executed successfully.

CL_RULE_RC(CL_ERR_NULL_POINTER): `pFirstExpr` or `pNextExpr` contains a NULL pointer.

Description:

This function appends an expression to another. It is used to create complex expressions by combining existing expressions.

Library File:

libClUtils

Related Function(s):

clRuleExprAllocate

3.2.4 clRuleExprDuplicate

clRuleExprDuplicate

Synopsis:

Duplicates a RBE expression.

Header File:

clRuleApi.h

Syntax:

```
ClRcT clRuleExprDuplicate(
            CL_IN ClRuleExprT* pSrcExpr,
            CL_OUT ClRuleExprT** ppDstExpr);
```

Parameters:

pSrcExpr: (in) Source RBE expression to Copy.

ppDstExpr: (out) Pointer to a new copy of the expression.

Return values:

CL_OK: The function executed successfully.

CL_RC_ERROR: An error has occurred.

Description:

This function makes a copy of the given RBE expression. The function allocates the required memory that must be freed by the caller using `clRuleExprDeallocate()`.

Library File:

libClUtils

Related Function(s):

clRuleExprAllocate, clRuleExprDeallocate

3.2.5 clRuleExprEvaluate

clRuleExprEvaluate

Synopsis:

Evaluates a complex RBE expression.

Header File:

clRuleApi.h

Syntax:

```
    ClRuleResultT  clRuleExprEvaluate(
                                CL_IN ClRuleExprT* pExpr,
                                CL_IN ClUint32T *pData,
                                CL_IN int dataLen);
```

Parameters:

pExpr: (in) RBE expression to be evaluated.

pData: (in) Data pointer against which the RBE needs to be compared.

dataLen: (in) Length of the data in multiples of 4 bytes.

Return values:

CL_RULE_TRUE: The RBE expression evaluates to `TRUE`.

CL_RULE_FALSE: The RBE expression evaluates to `FALSE`.

Description:

This function evaluates an RBE expression. RBE expression could be a complex expression, that is, multiple expressions can be chained together against a flat buffer `pData`, of length `dataLen`.

Library File:

libClUtils

Related Function(s):

clRuleDoubleExprEvaluate

3.2.6 clRuleDoubleExprEvaluate

clRuleDoubleExprEvaluate

Synopsis:

Evaluates Double RBE expressions.

Header File:

clRuleApi.h

Syntax:

```
ClRuleResultT  clRuleDoubleExprEvaluate
                            CL_IN ClRuleExprT* pExpr1,
                            CL_IN ClRuleExprT* pExpr2);
```

Parameters:

pExpr1: (in) First RBE expression to be evaluated.

pExpr2: (in) Second RBE expression to be evaluated.

dataLen: (in) Length of the data in multiples of 4 bytes.

Return values:

CL_RULE_TRUE: The RBE expression evaluates to TRUE.

CL_RULE_FALSE: The RBE expression evaluates to FALSE.

Description:

This function evaluates an RBE expression against another RBE expression. Both expressions are assumed to be of simple type.

Library File:

libClUtils

Related Function(s):

clRuleExprEvaluate

3.2.7 clRuleExprLocalConvert

clRuleExprLocalConvert

Synopsis:
Converts an RBE expression to local endianess.

Header File:
clRuleApi.h

Syntax:

```
    ClRcT clRuleExprLocalConvert(
                                 CL_IN ClRuleExprT* pExpr);
```

Parameters:
pExpr: (in) RBE expression to be converted.

Return values:
CL_OK: The function executed successfully.

CL_RC_ERROR: An error has occurred.

Description:
This function converts an RBE expression to match local endianess.

Library File:
libClUtils

Related Function(s):
clRuleExprConvert

3.2.8 clRuleExprConvert

clRuleExprConvert

Synopsis:

Endian converts a complex RBE expression.

Header File:

clRuleApi.h

Syntax:

```
    ClRcT clRuleExprConvert(
                                CL_IN ClRuleExprT* pExpr);
```

Parameters:

pExpr: (in) RBE expression to be converted.

Return values:

CL_OK: The function executed successfully.

CL_RC_ERROR: An error has occurred.

Description:

This function converts endianness of a complex RBE expression.

Library File:

libClUtils

Related Function(s):

clRuleExprLocalConvert

3.2.9 clRuleExprFlagsSet

clRuleExprFlagsSet

Synopsis:

Sets the flags of an RBE expression.

Header File:

clRuleApi.h

Syntax:

```
    ClRcT clRuleExprFlagsSet(
                            CL_IN ClRuleExprT* pExpr,
                            CL_IN ClRuleExprFlagsT flags);
```

Parameters:

pExpr: (in) RBE expression for which flags to be set.

flags: (in) Flags to be set.

Return values:

CL_OK: The function executed successfully.

CL_RC_ERROR: An error has occurred.

Description:

This function sets the RBE expression flags as specified by the parameter flags.

Library File:

libClUtils

Related Function(s):

clRuleExprFlagsGet

3.2.10 clRuleExprOffsetSet

clRuleExprOffsetSet

Synopsis:

Sets offset of a RBE expression.

Header File:

clRuleApi.h

Syntax:

```
    ClRcT clRuleExprOffsetSet (
                                  CL_IN ClRuleExprT* pExpr,
                                  CL_IN ClUint16T offset);
```

Parameters:

pExpr: (in) RBE expression for which flags to be set.

offset: (in) Offset value (multiples of 4 bytes).

Return values:

CL_OK: The function executed successfully.

CL_RC_ERROR: An error has occurred.

Description:

This function sets the offset field of an RBE expression as specified by the offset parameter.

Library File:

libClUtils

Related Function(s):

clRuleExprOffsetGet

3.2.11 clRuleExprMaskSet

clRuleExprMaskSet

Synopsis:

Set mask of an RBE expression.

Header File:

clRuleApi.h

Syntax:

```
    clRuleExprMaskSet (
                                CL_IN ClRuleExprT* pExpr,
                                CL_IN ClUint16T offset,
                                CL_IN ClUint32T mask);
```

Parameters:

pExpr: (in) RBE expression for which flags to be set.

offset: (in) Offset at which the mask to be set.

mask: (in) Mask value to be set.

Return values:

CL_OK: The function executed successfully.

CL_RC_ERROR: An error has occurred.

Description:

This function sets the mask field of an RBE expression as specified by the parameter `mask`.

Library File:

libClUtils

Note:

Offset is a multiple of 4 bytes and refers to the overall offset. `mask` is set to `expr->mask[offset - expr->offset]`.

Related Function(s):

clRuleExprMaskGet

3.2.12 clRuleExprValueSet

clRuleExprValueSet

Synopsis:
Sets value of an RBE expression.

Header File:
clRuleApi.h

Syntax:

```
    ClRcT clRuleExprValueSet (
                                CL_IN ClRuleExprT* pExpr,
                                CL_IN ClUint16T offset,
                                CL_IN ClUint32T value);
```

Parameters:

pExpr: (in) RBE expression for which flags are to be set.

offset: (in) Offset at which the value is to be set.

mask: (in) Mask value to be set.

Return values:

CL_OK: The function executed successfully.

CL_RC_ERROR: An error has occurred.

Description:
This function sets the value field of an RBE expression to the value specified by the parameter `value`.

Library File:
libClUtils

Related Function(s):
clRuleExprValueGet

3.2.13 clRuleExprFlagsGet

clRuleExprFlagsGet

Synopsis:
Retrieves RBE expression flags.

Header File:
clRuleApi.h

Syntax:

```
    ClRcT clRuleExprFlagsGet (
                                CL_IN ClRuleExprT* pExpr,
                                CL_OUT ClRuleExprFlagsT *pFlags);
```

Parameters:

pExpr: (in) RBE expression for which flags to be set.

pFlags: (out) Flags to be returned.

Return values:

CL_OK: The function executed successfully.

CL_RC_ERROR: An error has occurred.

Description:
This function retrieves the flags of an RBE expression.

Library File:
libClUtils

Related Function(s):
clRuleExprFlagsGet

3.2.14 clRuleExprOffsetGet

clRuleExprOffsetGet

Synopsis:
Retrieves RBE expression Offset value.

Header File:
clRuleApi.h

Syntax:

```
ClRcT clRuleExprOffsetGet (
                           CL_IN ClRuleExprT* pExpr,
                           CL_OUT ClUint16T *pOffset);
```

Parameters:

pExpr: (in) RBE expression for which flags to be set.

pOffset: (out) Offset to be returned.

Return values:

CL_OK: The function executed successfully.

CL_RC_ERROR: An error has occurred.

Description:
This function retrieves the offset value of an RBE expression that is set using `clRuleExprOffsetSet()`.

Library File:
libClUtils

Related Function(s):
clRuleExprOffsetSet

3.2.15 clRuleExprMaskGet

clRuleExprMaskGet

Synopsis:

Retrieves RBE expression mask value.

Header File:

clRuleApi.h

Syntax:

```
    ClRcT clRuleExprMaskGet (
                            CL_IN ClRuleExprT* pExpr,
                            CL_IN ClUint16T offset,
                            CL_OUT ClUint32T *pMask);
```

Parameters:

pExpr: (in) RBE expression for which flags to be set.

offset: (in) Get Mask from this offset.

pMask: (out) Mask to be returned.

Return values:

CL_OK: The function executed successfully.

CL_RC_ERROR: An error has occurred.

Description:

This function retrieves the mask of an RBE expression that is set using `clRuleExprMaskSet()`.

Library File:

libClUtils

Related Function(s):

clRuleExprMaskSet

3.2.16 clRuleExprValueGet

clRuleExprValueGet

Synopsis:

Retrieves RBE expression value.

Header File:

clRuleApi.h

Syntax:

```
    ClRcT clRuleExprValueGet (
                                CL_IN ClRuleExprT* pExpr,
                                CL_IN ClUint16T offset,
                                CL_OUT ClUint32T *pValue);
```

Parameters:

pExpr :(in) RBE expression for which flags to be set.

offset :(in) Get mask from this offset.

pValue :(out) Value to be returned.

Return values:

CL_OK: The function executed successfully.

CL_RC_ERROR: An error has occurred.

Description:

This function retrieves the value of an RBE expression that is set using `clRuleExprValueSet()`.

Library File:

libClUtils

Related Function(s):

clRuleExprValueSet

3.2.17 clRuleExprMemLenGet

clRuleExprMemLenGet

Synopsis:

Retrieves the total memory used by the expression.

Header File:

clRuleApi.h

Syntax:

```
    ClUint32T clRuleExprMemLenGet (
                                    CL_IN ClRuleExprT* pExpr);
```

Parameters:

pExpr (in) RBE expression.

Return Value:

Length in bytes needed to pack an expression.

Description:

This function returns the memory needed to pack an expression into a contiguous memory location.

Library File:

libClUtils

Related Function(s):

clRuleExprPack

3.2.18 clRuleExprPack

clRuleExprPack

Synopsis:
Packs an RBE expression into the given memory area.

Header File:
clRuleApi.h

Syntax:

```
    ClRcT clRuleExprPack (
                            CL_IN ClRuleExprT* pSrcExpr,
                            CL_IN char *pBuf);
```

Parameters:

pSrcExpr: (in) Source RBE expression to pack.

pBuf: (in) Pointer to the memory location where to pack.

Return values:

CL_OK: The function executed successfully.

CL_RULE_RC(CL_ERR_NULL_POINTER): `pSrcExpr` or `pBuf` contains a NULL pointer.

Description:
This function packs the given RBE expression. The expression is packed at the given memory pointed by `pBuf`. It is assumed that the `pBuf` contains enough space to accommodate `srcExpr`. The caller can call `clRuleExprMemLenGet()` to obtain the memory needed to pack `srcExpr`.

Library File:
libClUtils

Related Function(s):
clRuleExprMemLenGet, clRuleExprUnpack

3.2.19 clRuleExprUnpack

clRuleExprUnpack

Synopsis:

Unpacks an RBE expression.

Header File:

clRuleApi.h

Syntax:

```
    ClRcT clRuleExprUnpack(
                            CL_IN ClRuleExprT* pSrcExpr,
                            CL_OUT ClRuleExprT** ppDstExpr);
```

Parameters:

pSrcExpr: (in) Source RBE expression to unpack.

pDstExpr: (out) Pointer to a new copy of the expression.

Return values:

CL_OK: The function executed successfully.

CL_RULE_RC(CL_ERR_NULL_POINTER): `pSrcExpr` or `ppDstExpr` contains a NULL pointer.

Description:

This function unpacks the given RBE expression.

Library File:

libClUtils

Related Function(s):

clRuleExprPack

3.2.20 clRuleExprPrint

clRuleExprPrint

Synopsis:
Prints a complex RBE expression.

Header File:
clRuleApi.h

Syntax:

```
ClRcT clRuleExprPrint (
                            CL_IN ClRuleExprT* pExpr);
```

Parameters:
pExpr: (in) RBE expression to be printed.

Return Value:
None.

Description:
This function prints the given RBE expression.

Library File:
libClUtils

Related Function(s):
None.

Chapter 4

Service Management Information Model

TBD

Chapter 5

Service Notifications

TBD

Chapter 6

Debug CLIs

TBD

Index