

ANDROID Description

Signatures for getting a logit.

```
public Logit getLogit( Class clazz )
public Logit getLogit( String logitName )
```

where:

- clazz - a Java class which uses the logit. All logit messages will use the type name.
- logitName - any string value which identifies the logit.

Signatures for logging a message/exception.

(Methods available in the Application.API.Logging.Logit class.)

```
public void debug( String message )
public void info( String message )
public void warn( String message )
public void warn( String message, Throwable t )
public void error( String message )
public void error( String message, Throwable t )
public void fatal( String message )
public void fatal( String message, Throwable t )
public void trace( String message )
```

where:

- message - the message to log.
- t - an exception or error to log. Application logs a stack trace for the exception.

Signatures for configuring log buffer policy.

(The method is available via the application logging accessory.)

```
public void setLogitReporterPolicy( int numOfMessages, int timeFrequencyMS )
```

where:

- numOfMessages - the upper limit of the number of messages. Excess invokes buffer flush method after specified time.
- timeFrequencyMS - time frequency/interval in milliseconds defining how often log messages from the buffer should be flushed to the server.

Logit to submit an info log level message.

```
Logit logit = Application.Logging.getLogit( "com.mbaas.Logit" );
logit.info( "Starting application.." );
```

Configuring a log buffering policy for 100 messages and flush delay of 60 seconds.

```
Application.Logging.setLogitReporterPolicy(100, 60000);
```

iOS comment : 0222201350929 : user edit

OBJ-C:

Signatures for Retrieving a Logit

```
-(Logit *)getLogitClass:(Class)clazz;
-(Logit *)getLogit:(OBString *)logitName;
```

where:

- clazz - a class which uses the logit. All logit messages will use the type name
- logitName - any string value which identifies the logit.

Signatures for logging a message/exception.

(Methods available in the Logit class.)

```
-(void)debug:(OBString *)message;
-(void)info:(OBString *)message;
-(void)trace:(OBString *)message;
-(void)warn:(OBString *)message;
-(void)warn:(OBString *)message exception:(OBException *)exception;
-(void)error:(OBString *)message;
-(void)error:(OBString *)message exception:(OBException *)exception;
-(void)fatal:(OBString *)message;
-(void)fatal:(OBString *)message exception:(OBException *)exception;
```

where:

- message - the message to log;
- exception- an exception or error to log. Application logs a stack trace for the exception.