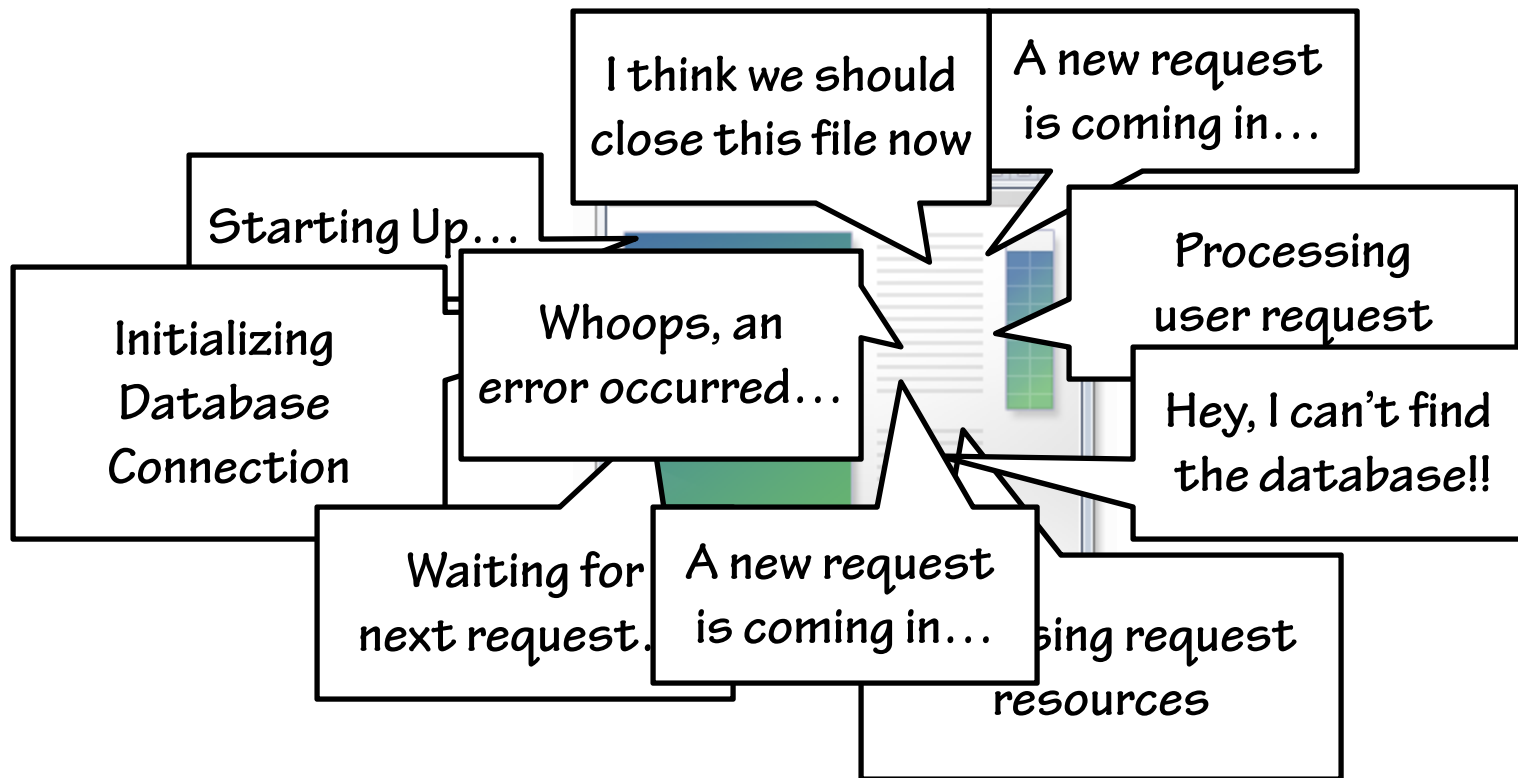
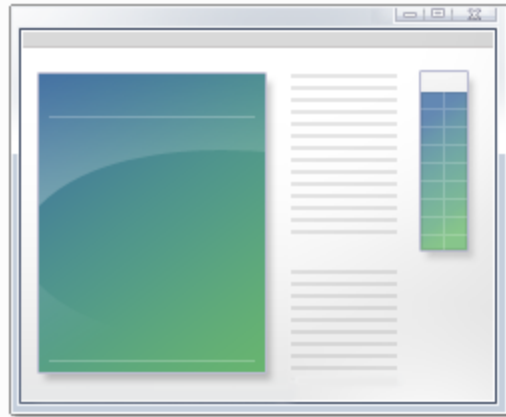


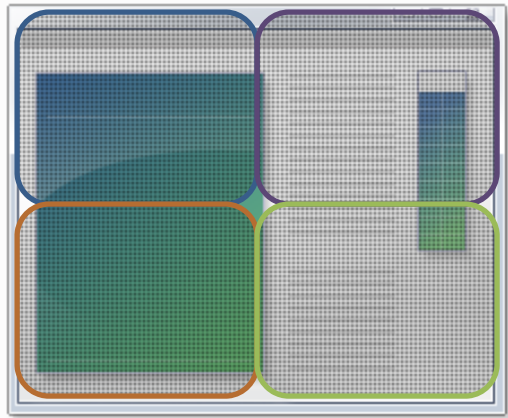
Logger Objects

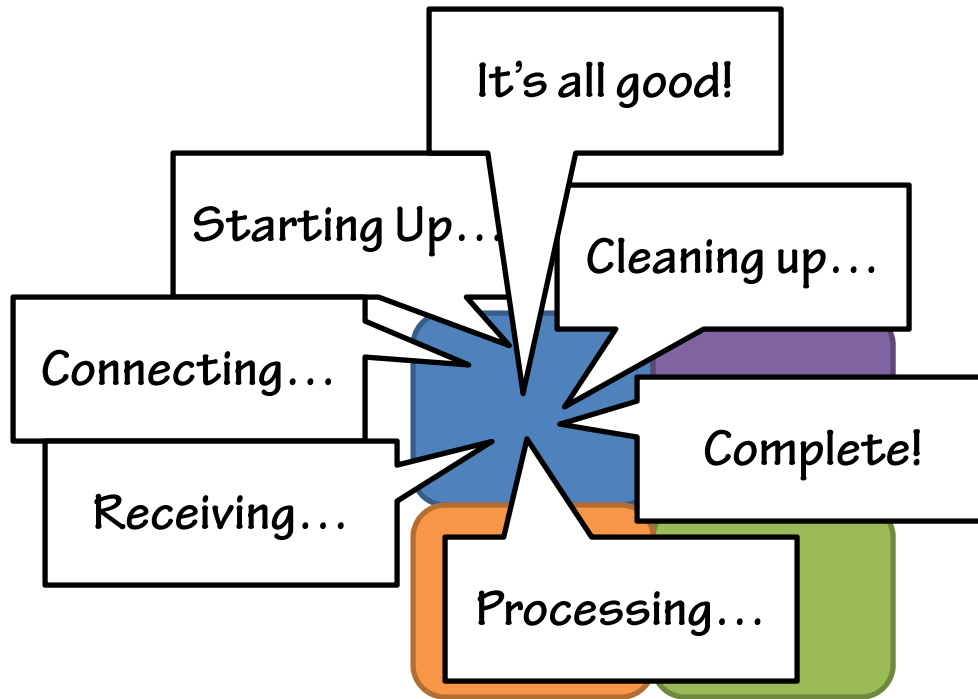
Jim Christopher
@beefarino

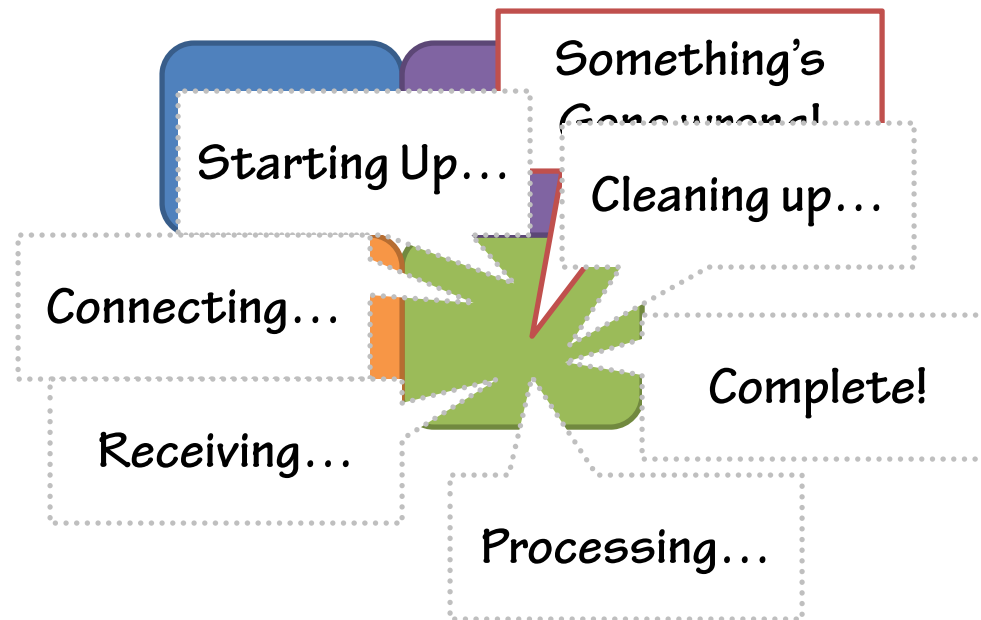




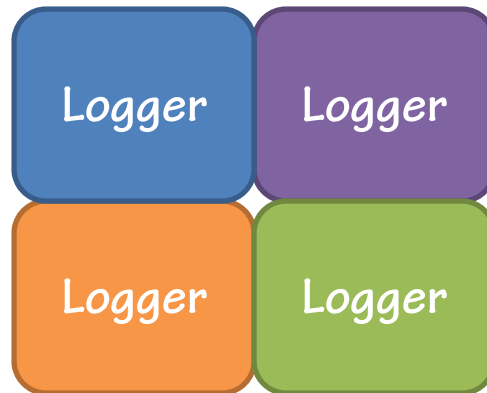


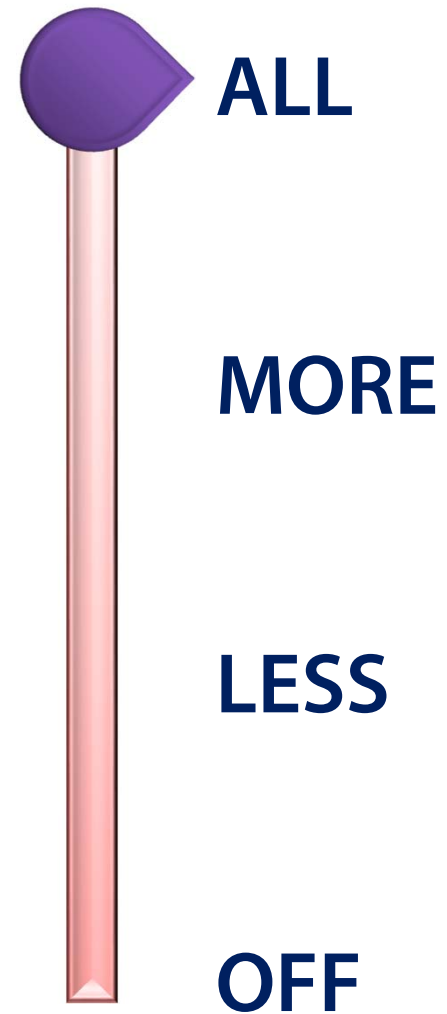
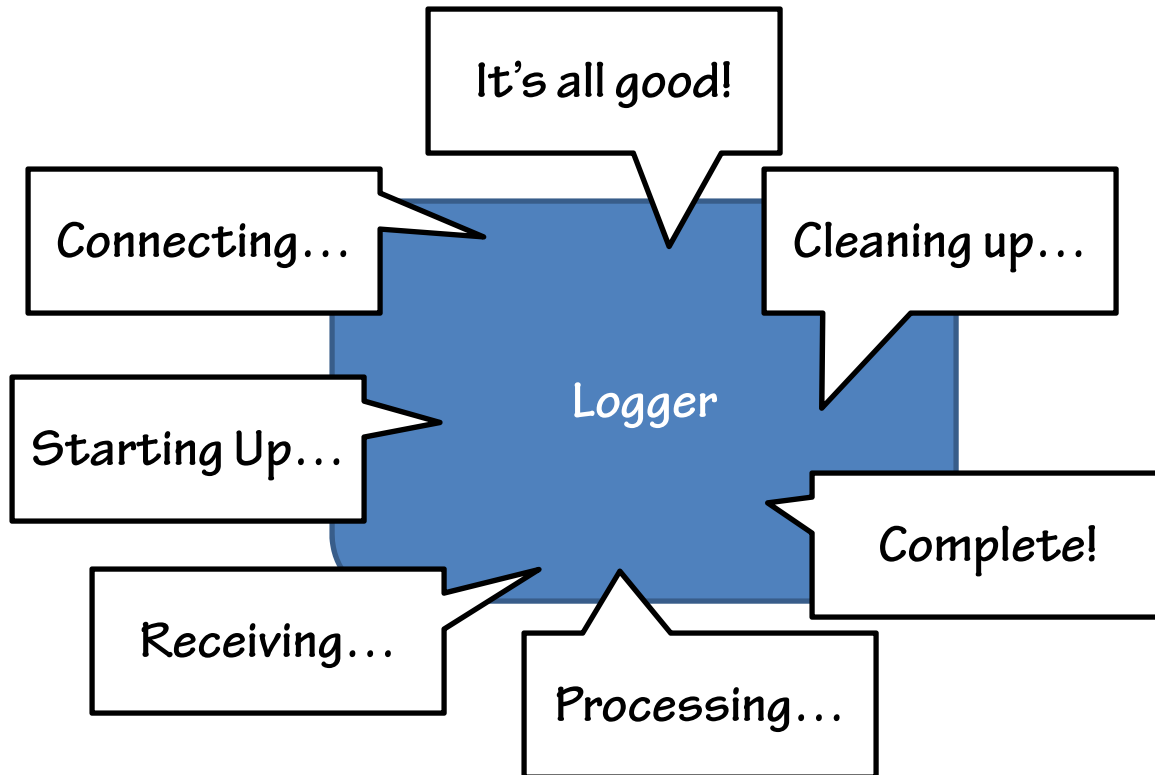


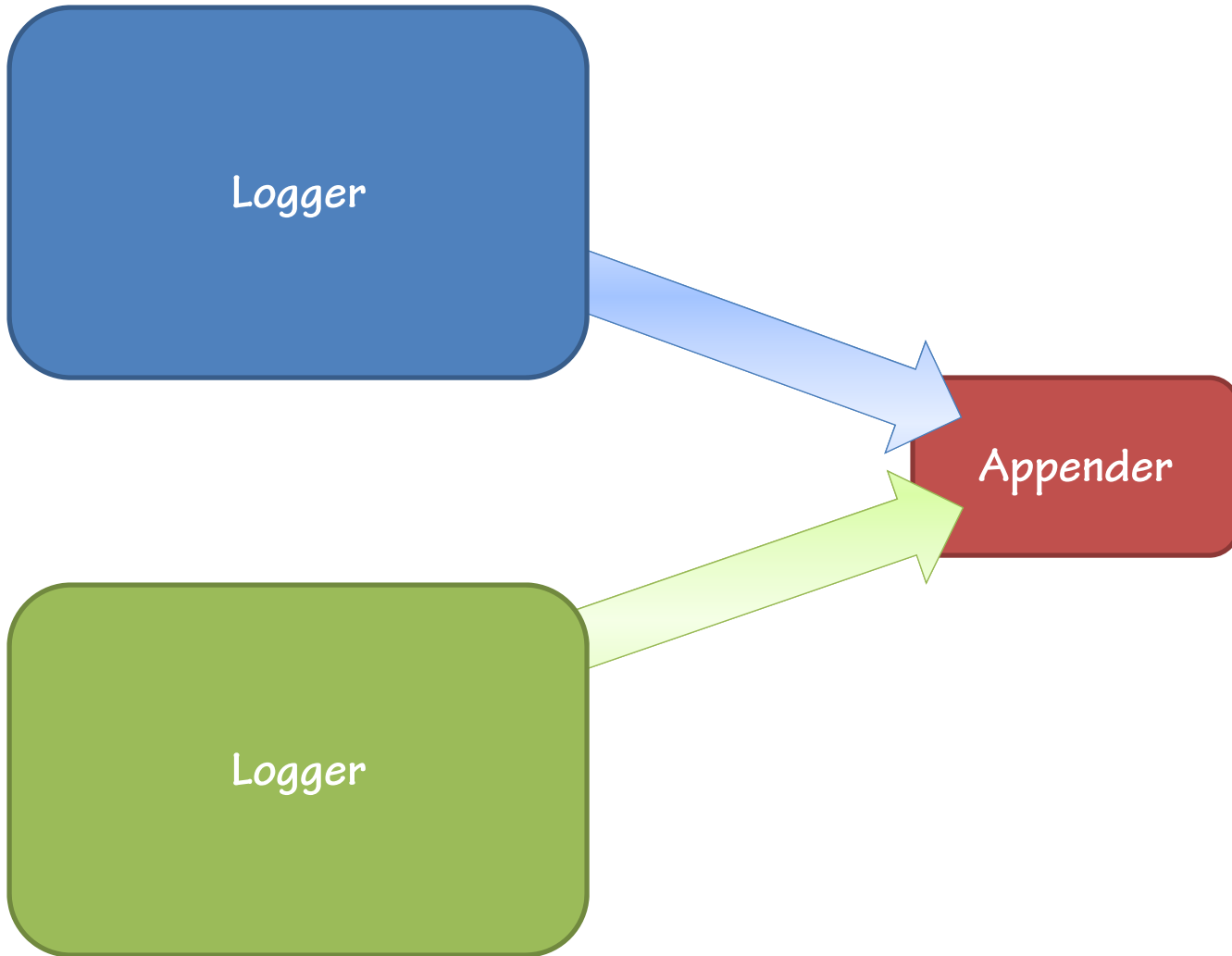


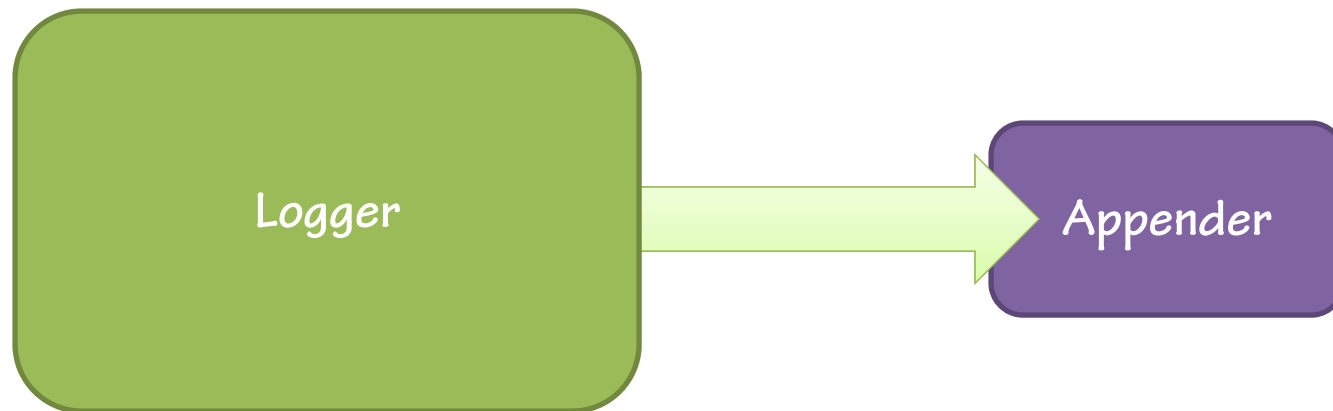
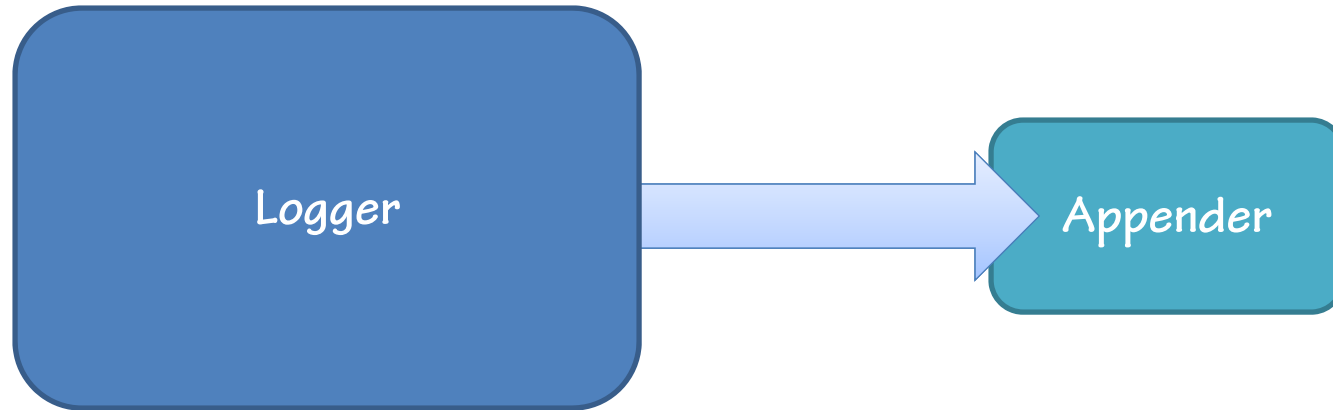


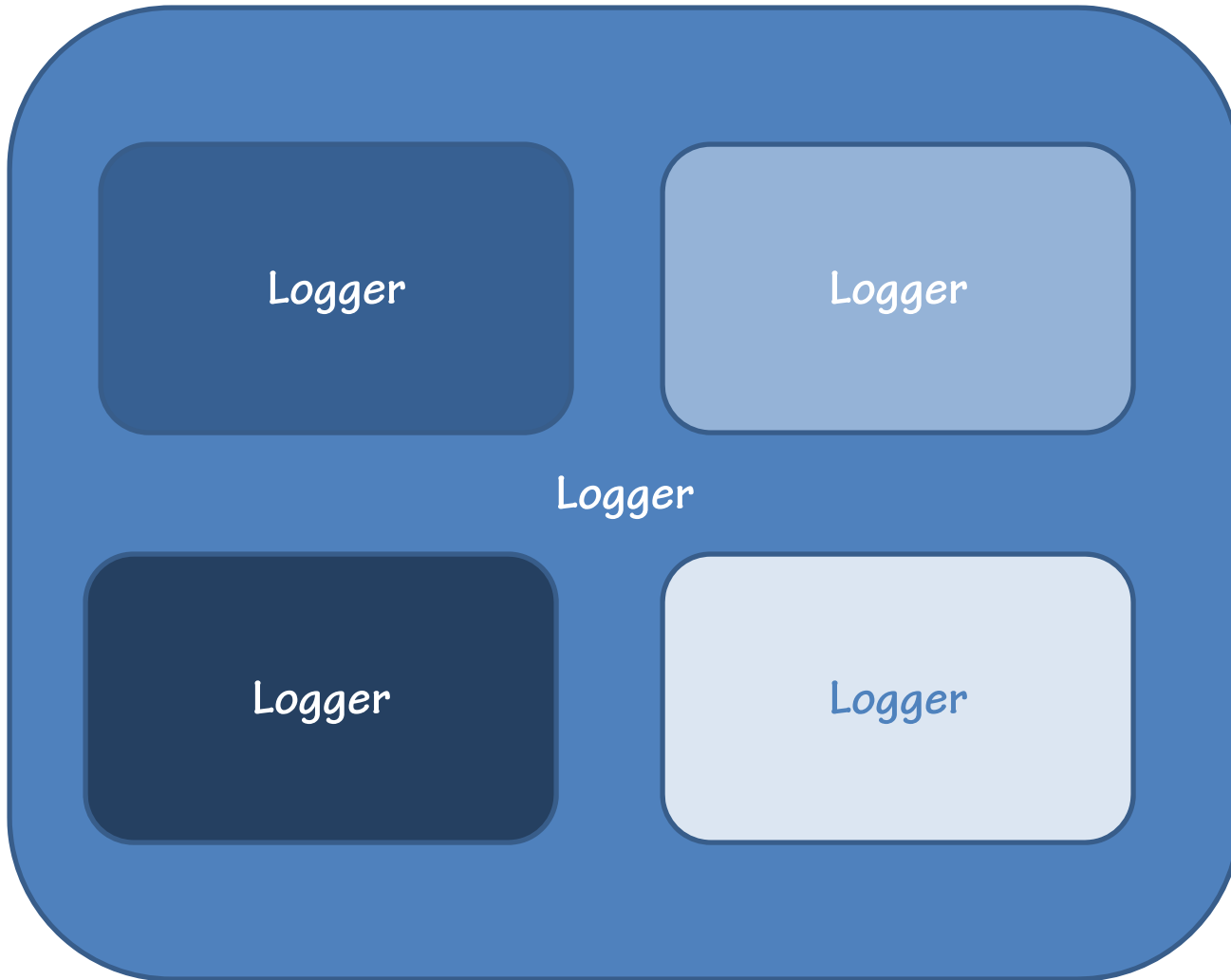


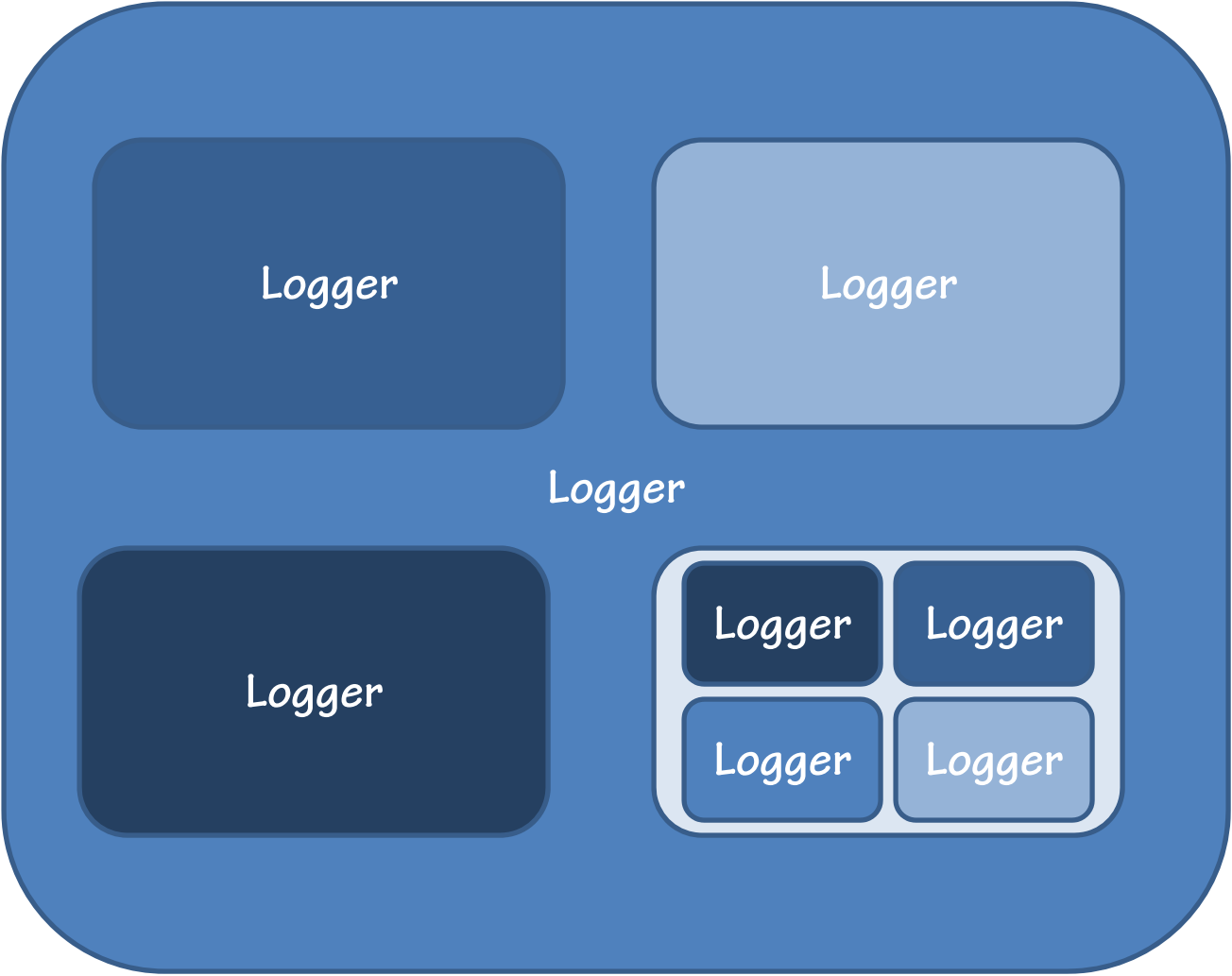














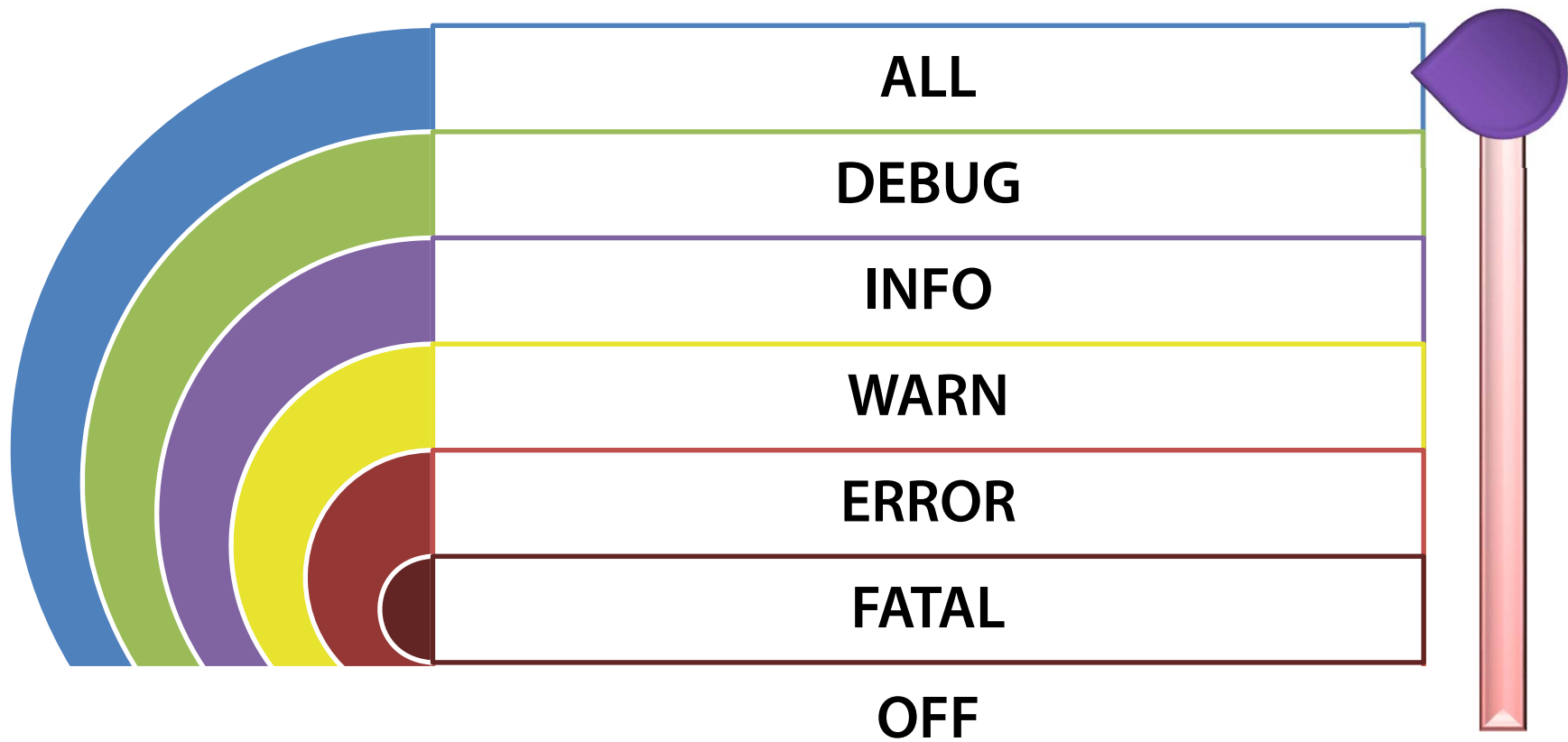
DEBUG

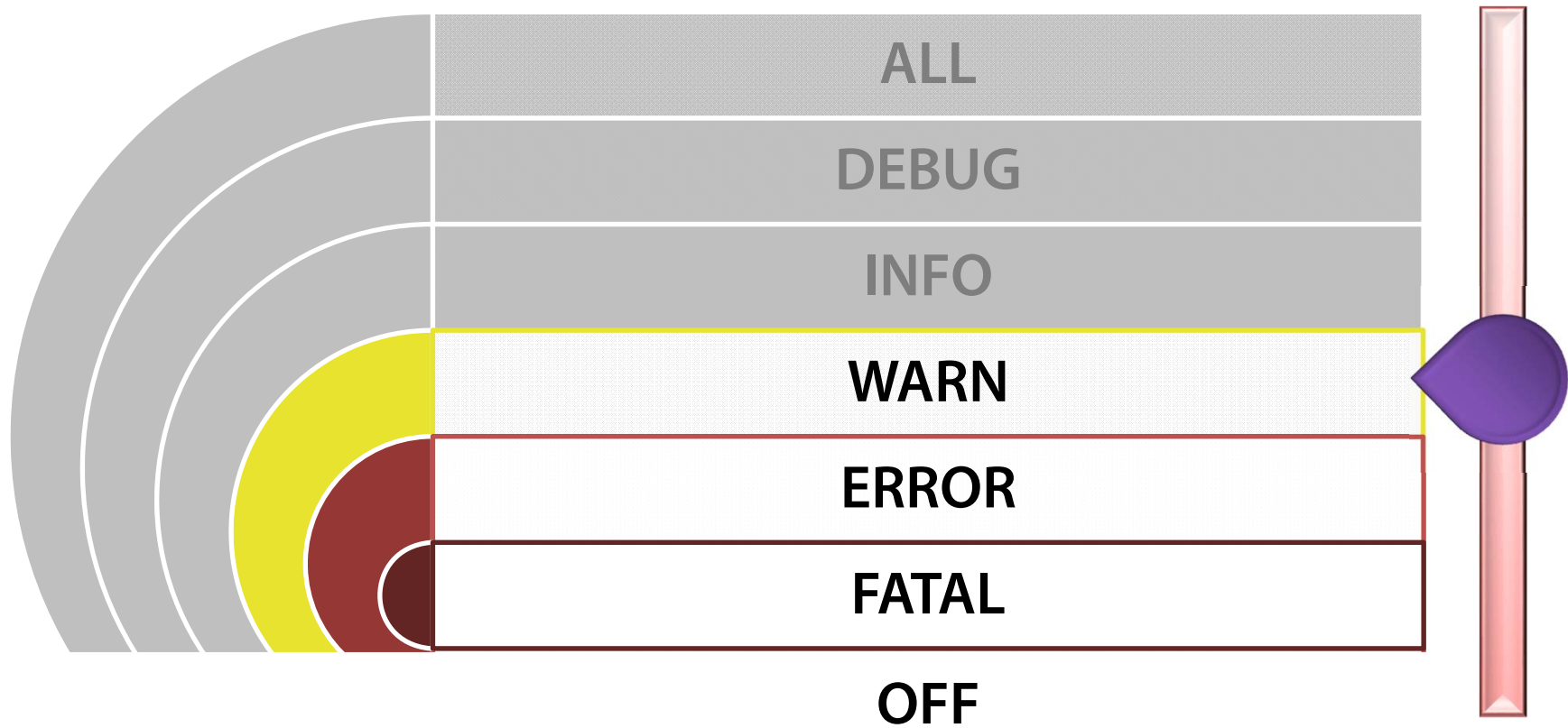
INFO

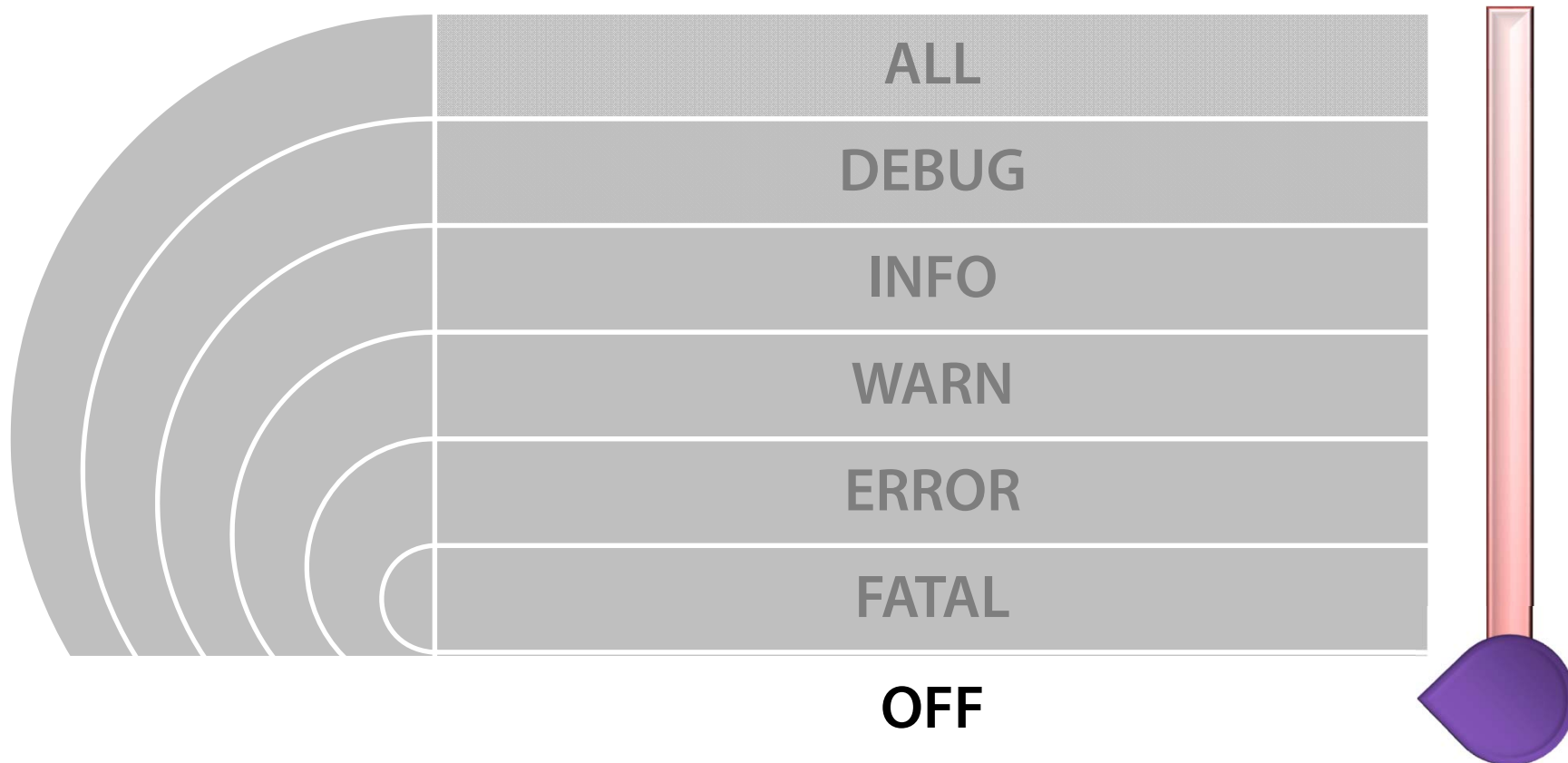
WARN

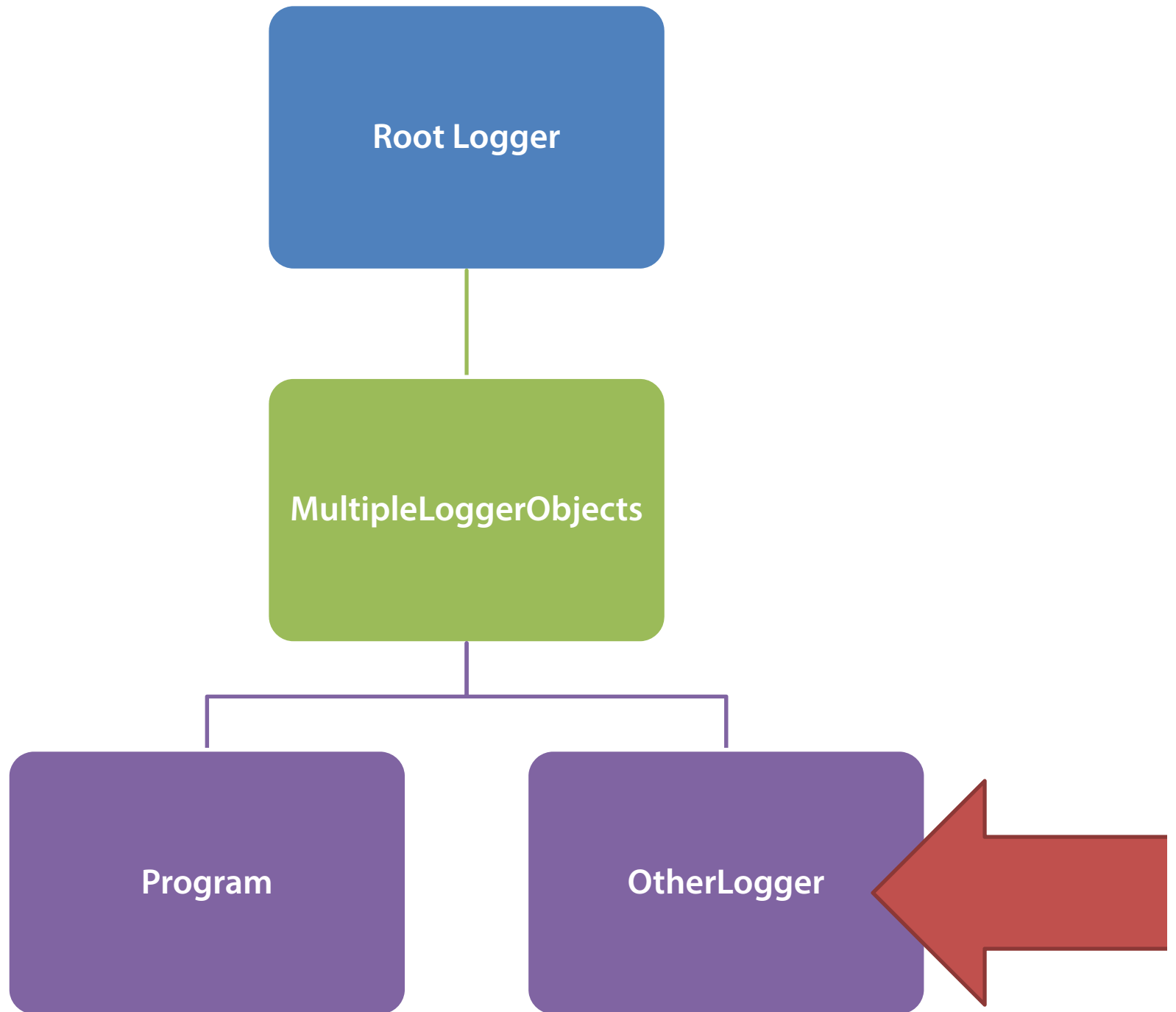
ERROR

FATAL







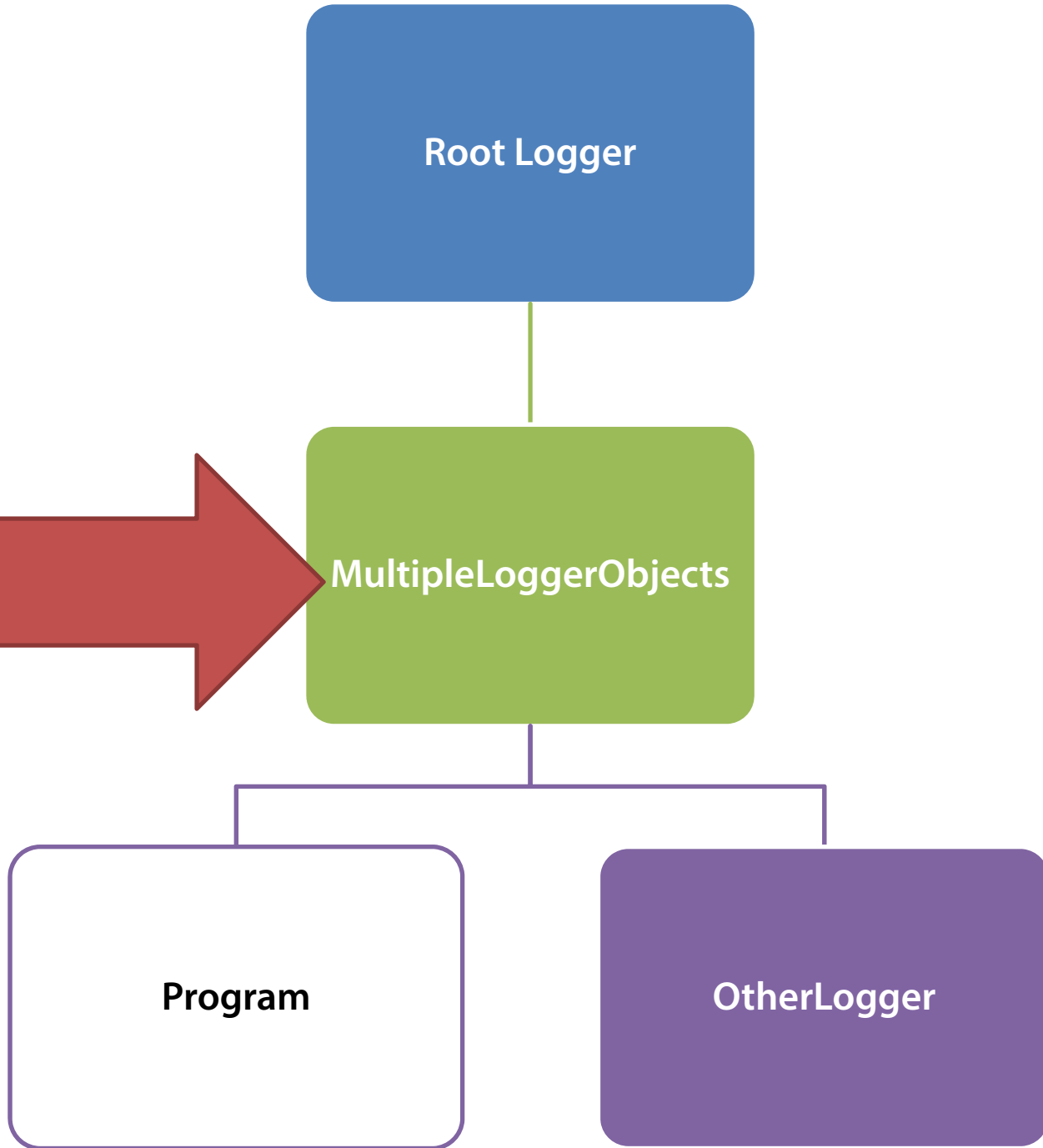


Root Logger

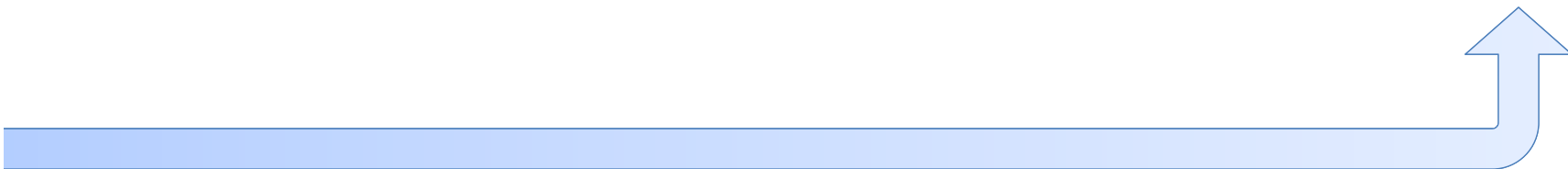
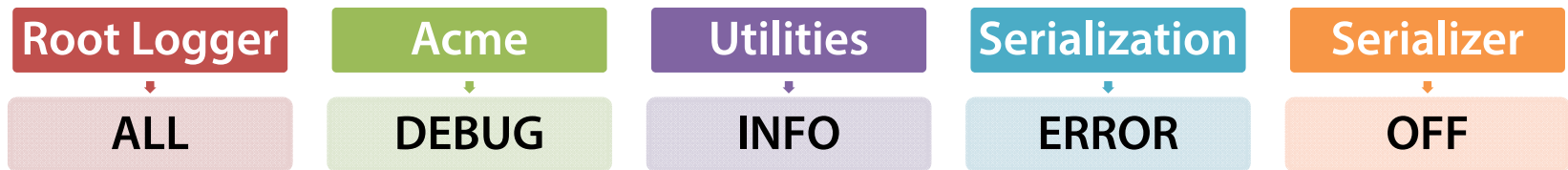
MultipleLoggerObjects

Program

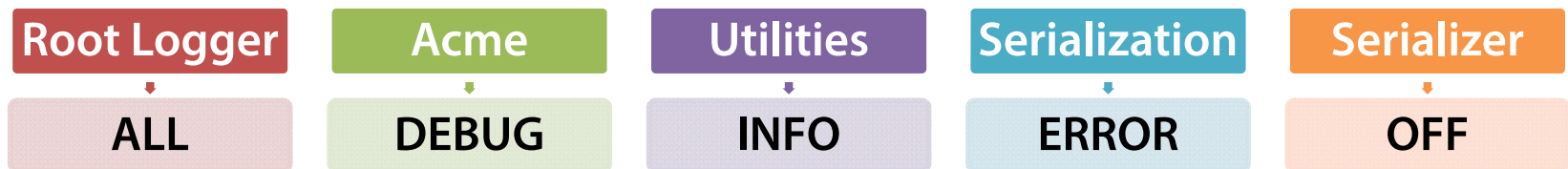
OtherLogger



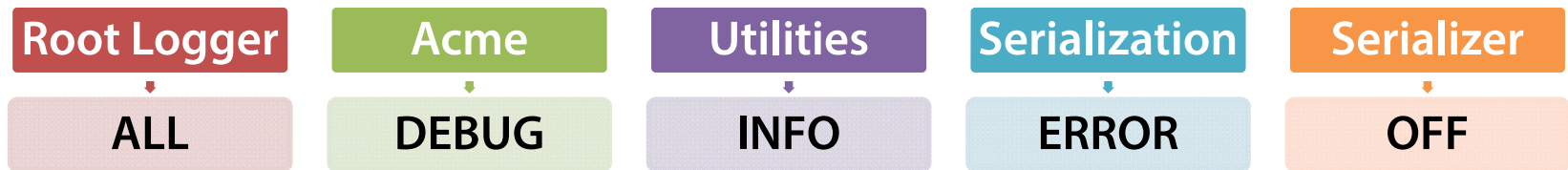
```
var type = typeof( Acme.Utilities.Serialization.Serializer );  
var logger = log4net.LogManager.GetLogger(type);  
logger.Info( "This is a log message!" );
```



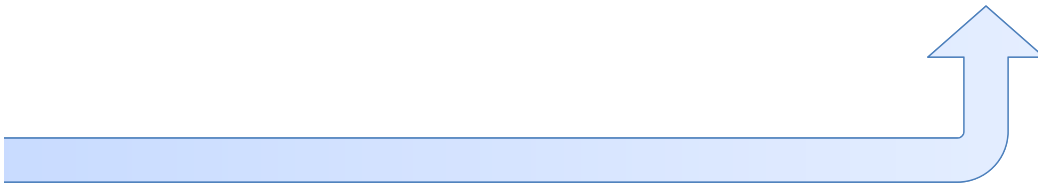
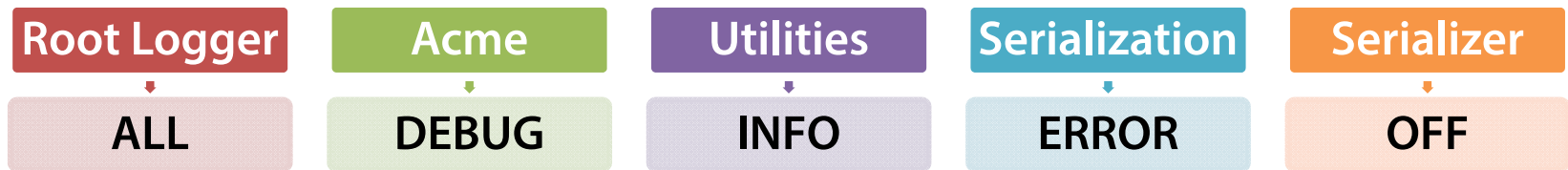
```
var type = typeof( Acme.Utilities.Serialization.Exception );  
var logger = log4net.LogManager.GetLogger(type);  
logger.Info( "This is a log message!" );
```



```
var type = typeof( Acme.Utilities.Streams.Extensions );  
var logger = log4net.LogManager.GetLogger(type);  
logger.Info( "This is a log message!" );
```



```
var type = typeof( Foobar.Kenooteroid );  
var logger = log4net.LogManager.GetLogger(type);  
logger.Info( "This is a log message!" );
```



Appenders

Accumulate

Appender

Root Logger

Appender

Acme

Appender

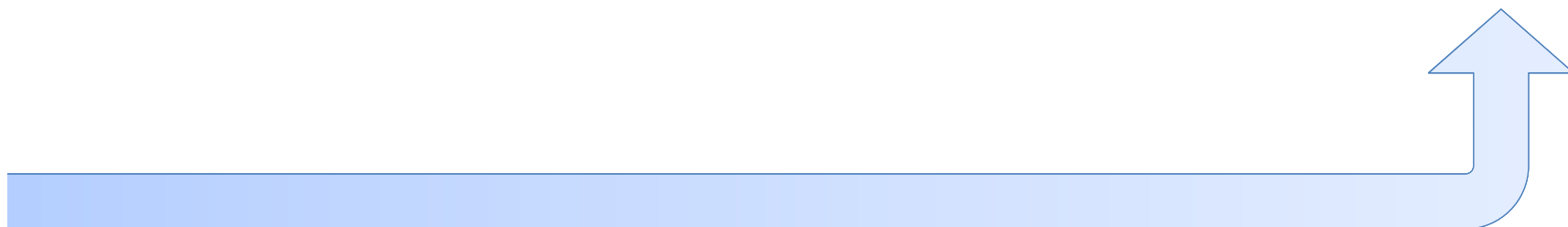
Utilities

Appender

Serialization

Appender

Serializer



Logger Objects

- **Create Logger objects**
 - `log4net.LogManager.GetLogger` method

Logger Objects

- Create Logger objects
 - `log4net.LogManager.GetLogger` method
- **log4net.ILog interface**
 - Log raw objects
 - Formatting methods
 - Conditional logging properties

Logger Objects

- Create Logger objects
 - log4net.LogManager.GetLogger method
- log4net.ILog interface
 - Log raw objects
 - Formatting methods
 - Conditional logging properties
- Control logger verbosity
 - Level property
 - ALL = DEBUG > INFO > WARN > ERROR > FATAL > OFF

Logger Objects

- **Create Logger objects**
 - log4net.LogManager.GetLogger method
- **log4net.ILog interface**
 - Log raw objects
 - Formatting methods
 - Conditional logging properties
- **Control logger verbosity**
 - Level property
 - ALL = DEBUG > INFO > WARN > ERROR > FATAL > OFF
- **Common logger object naming convention**
 - Full name of the .NET type
 - Logger hierarchy

Logger Objects

- **Create Logger objects**
 - log4net.LogManager.GetLogger method
- **log4net.ILog interface**
 - Log raw objects
 - Formatting methods
 - Conditional logging properties
- **Control logger verbosity**
 - Level property
 - ALL = DEBUG > INFO > WARN > ERROR > FATAL > OFF
- **Common logger object naming convention**
 - Full name of the .NET type
 - Logger hierarchy
- **Attach appenders to loggers**
 - Appenders accumulate across the logger hierarchy