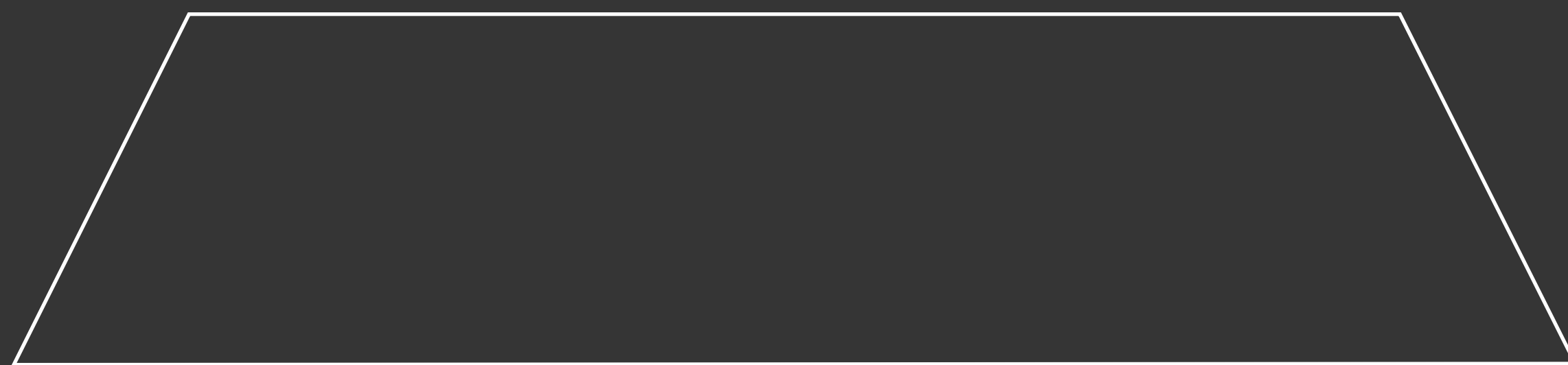


A solid teal vertical bar is positioned on the far left side of the image.

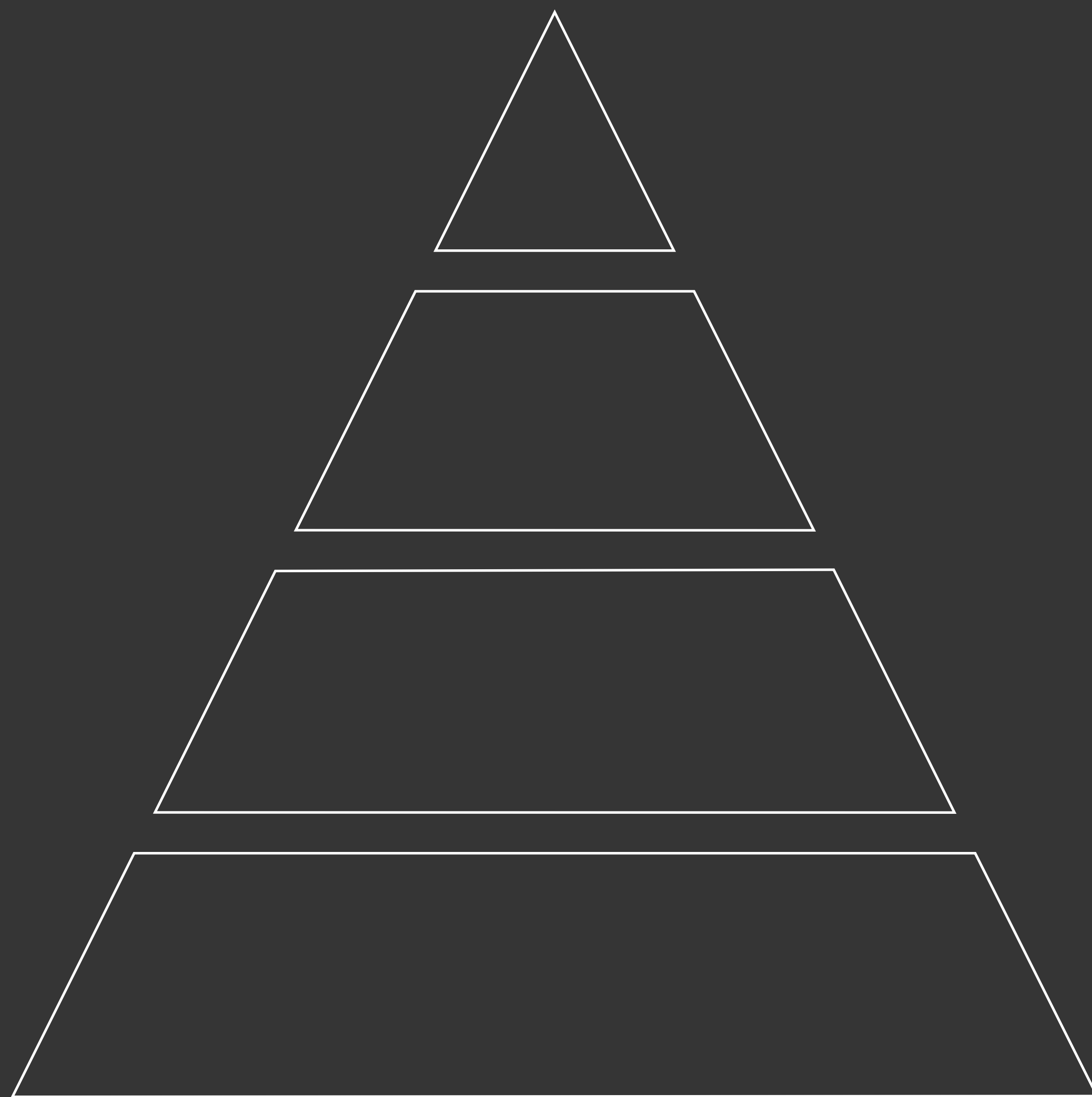
okio



okio



okio



retrofit

moshi

okhttp

okio

okio?

HTTP

HTTP

```
GET /a-few-ok-libraries.html HTTP/1.1  
Host: droidcon.ca
```


HTTP

```
GET /a-few-ok-libraries.html HTTP/1.1  
Host: droidcon.ca
```

```
HTTP/1.1 200 OK  
Date: Thu, 9 Apr 2015 15:14:34 GMT  
Server: JakeWharton/2.3.7 (Human)  
Content-Type: application/presentation; charset=UTF-8  
Content-Length: 3000
```

...

HTTP

HTTP/1.1 200 OK

Date: Thu, 9 Apr 2015 15:14:34 GMT

Server: JakeWharton/2.3.7 (Human)

Content-Type: application/presentation; charset=UTF-8

Content-Length: 3000

...

HTTP

HTTP/1.1 · 200 · OK\r\n

Date: · Thu, · 9 · Apr · 2015 · 15:14:34 · GMT\r\n

Server: · JakeWharton/2.3.7 · (Human)\r\n

Content-Type: · application/presentation; · charset=UTF-8\r\n

Content-Length: · 3000\r\n

\r\n

...

HTTP

HTTP/1.1·200·OK\r\n

Date:·Thu,·9·Apr·2015·15:14:34·GMT\r\n

Server:·JakeWharton/2.3.7·(Human)\r\n

Content-Type:·application/presentation;·charset=UTF-8\r\n

Content-Length:·3000\r\n

\r\n

...

HTTP

```
HTTP/1.1·200·OK\r\nDate:·Thu,·9·Apr·2015·15:14:34·GMT\r\nServer:·JakeWharto  
n/2.3.7·(Huma  
n)\r\nContent-Type:·application/presentation;·charset  
=UTF-8\r\nContent-Length:·3000\r  
\n\r\n...
```

HTTP

```
HTTP/1.1·200·OK\r\nDate:·Thu,·9·Apr·2015·15:14:34·GMT\r\nServer:·JakeWharto  
n/2.3.7·(Huma  
n)\r\nContent-Type:·application/presentation;·charset  
=UTF-8\r\nContent-Length:·3000\r  
\n\r\n...
```

HTTP

```
HTTP/1.1·200·OK\r\nDate:·Thu,·9·Apr·2015·15:14:34·GMT\r\nServer:·JakeWharto  
n/2.3.7·(Huma  
n)\r\nContent-Type:·application/presentation;·charset  
=UTF-8\r\nContent-Length:·3000\r  
\n\r\n...
```

HTTP

```
      Date: • Tue, 9 Apr 2015 15:14:34 GMT\r\nServer: • JakeWharto  
n/2.3.7 • (Huma  
n)\r\nContent-Type: • application/presentation; charset  
=UTF-8\r\nContent-Length: • 3000\r  
\n\r\n...
```

HTTP/1.1 200 OK

HTTP

```
      Date: ·T
hu, ·9·Apr·2015·15:14:34·GMT\r\nServer: ·JakeWharto
n/2.3.7·(Huma
n)\r\nContent-Type: ·application/presentation; ·charset
=UTF-8\r\nContent-Length: ·3000\r
\r\n\r\n...
```

HTTP/1.1 200 OK

HTTP

```
      Date: • Tue, 9 Apr 2015 15:14:34 GMT\r\nServer: • JakeWharto  
n/2.3.7 • (Huma  
n)\r\nContent-Type: • application/presentation; charset  
=UTF-8\r\nContent-Length: • 3000\r  
\n\r\n...
```

HTTP/1.1 200 OK

HTTP

Server: · JakeWharto

n/2.3.7· (Huma
n)\r\nContent-Type: · application/presentation; · charset
=UTF-8\r\nContent-Length: · 3000\r
\n\r\n...

HTTP/1.1 200 OK

Date Thu, 9 Apr 2015 15:14:34 GMT

java.io.*

java.io.*

```
abstract class InputStream {  
}
```

java.io.*


```
abstract class InputStream {  
    void close();  
}
```

java.io.*

```
abstract class InputStream {  
    void close();  
    long skip(long byteCount);  
}
```

java.io.*

```
abstract class InputStream {  
    void close();  
    long skip(long byteCount);  
}
```



java.io.*

```
abstract class InputStream {  
    void close();  
    long skip(long byteCount);  
}
```

```
while (byteCount > 0) {  
    byteCount -= is.skip(byteCount);  
}
```

java.io.*

```
abstract class InputStream {  
    void close();  
    long skip(long byteCount);  
}
```

```
public static void reallySkip(InputStream is, long byteCount) {  
    while (byteCount > 0) {  
        byteCount -= is.skip(byteCount);  
    }  
}
```

java.io.*

```
abstract class InputStream {  
    void close();  
    int read(byte[] buffer);  
    int read(byte[] buffer, int offset, int count);  
    long skip(long byteCount);  
}
```

java.io.*

```
abstract class InputStream {  
    void close();  
    int read();  
    int read(byte[] buffer);  
    int read(byte[] buffer, int offset, int count);  
    long skip(long byteCount);  
}
```

java.io.*

```
abstract class InputStream {  
    int available();  
    void close();  
    void mark(int readLimit);  
    boolean markSupported();  
    int read();  
    int read(byte[] buffer);  
    int read(byte[] buffer, int offset, int count);  
    synchronized void reset();  
    long skip(long byteCount);  
}
```

java.io.*

- API is conflated with multiple concepts and inconsistent behavior

HTTP

Server: · JakeWharto

n/2.3.7· (Huma
n)\r\nContent-Type: · application/presentation; · charset
=UTF-8\r\nContent-Length: · 3000\r
\n\r\n...

HTTP/1.1 200 OK

Date Thu, 9 Apr 2015 15:14:34 GMT

HTTP

Server: · JakeWharto

n/2.3.7· (Huma
n)\r\nContent-Type: · application/presentation; · charset
=UTF-8\r\nContent-Length: · 3000\r
\n\r\n...

H	T	T	P	/	1	.	1	.	2	0	0	.	0	K	r	n	D	a	t	e	:	.	T	h	u	,	.	9	.	A	p	r	.	2	0	1	5	.	1	5	:	1	4	:	3	4	.	G	M	T	r	n
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

HTTP

Server: · JakeWharto

n/2.3.7· (Huma
n)\r\nContent-Type: · application/presentation; · charset
=UTF-8\r\nContent-Length: · 3000\r
\n\r\n...

HTTP/1.1 · 200 · OK\r\nDate: · Thu, · 9 · Apr · 2015 · 15:14:34 · GMT\r\n

HTTP

Server: · JakeWharto

n/2.3.7· (Huma
n)\r\nContent-Type: · application/presentation; · charset
=UTF-8\r\nContent-Length: · 3000\r
\n\r\n...

HTTP

```
n/2.3.7 (Huma
n)\r\nContent-Type: application/presentation; charset
=UTF-8\r\nContent-Length: 3000\r
\n\r\n...
```

```
Server: JakeWharto
```

HTTP

```
n)\r\nContent-Type: ·application/presentation; ·charset
=UTF-8\r\nContent-Length: ·3000\r
\r\n\r\n...
```

```
Server: ·JakeWharton/2.3.7· (Huma
```

HTTP

```
          lication/presentation; charset
=UTF-8\r\nContent-Length: 3000\r
\r\n...
```

```
Server: JakeWharton/2.3.7 (Human)\r\nContent-Type: app
```

HTTP

```
          lication/presentation; charset
=UTF-8\r\nContent-Length: 3000\r
\r\n...
```

```
Server: JakeWharton/2.3.7 (Human)\r\nContent-Type: app
```

HTTP

```
          lication/presentation; charset
=UTF-8\r\nContent-Length: 3000\r
\r\n...
```

Content-Type: app

Server JakeWharton/2.3.7 (Human)

HTTP

```
lication/presentation; charset
=UTF-8\r\nContent-Length: 3000\r
\n\r\n...
```

Content-Type: app

HTTP

```
lication/presentation; charset
=UTF-8\r\nContent-Length: 3000\r
\r\n...
```

Content-Type: app

HTTP

```
          lication/presentation; charset
=UTF-8\r\nContent-Length: 3000\r
\r\n...
```

```
Content-Type: app
```

```
Content-Type: app
```

HTTP

```
          lication/presentation; charset
=UTF-8\r\nContent-Length: 3000\r
\r\n...
```

```
Content-Type: app
```

HTTP

```
          lication/presentation; charset
=UTF-8\r\nContent-Length: 3000\r
\r\n...
```

```
Content-Type: app
```

HTTP

```
          lication/presentation; charset
=UTF-8\r\nContent-Length: 3000\r
\r\n...
```

HTTP

```
lication/presentation; charset  
=UTF-8\r\nContent-Length: 3000\r  
\n\r\n...
```

Content-Type: app

HTTP

lication/presentation; charset
=UTF-8\r\nContent-Length: 3000\r
\n\r\n...

Content-Type: app



HTTP

lication/presentation; charset
=UTF-8\r\nContent-Length: 3000\r
\n\r\n...

Content-Type: app



HTTP

=UTF-8\r\nContent-Length: 3000\r\n\r\n...

lication/presentation; charset Content-Type: app



HTTP

8\r\nContent-Length: 3000\r\n\r\n...

lication/presentation; charset=UTF-Content-Type: app



java.io.*

- API is conflated with multiple concepts and inconsistent behavior

java.io.*

- API is conflated with multiple concepts and inconsistent behavior
- Burden of figuring out storage is on the consumer

java.io.*

```
class BufferedInputStream extends InputStream {  
    BufferedInputStream(InputStream is) {  
        // ...  
    }  
  
    BufferedInputStream(InputStream is, int size) {  
        // ...  
    }  
}
```

java.io.*

```
n)\r\nContent-Type: ·application/presentation; ·charset  
=UTF-8\r\nContent-Length: ·3000\r  
\n\r\n...
```

Server: ·JakeWharton/2.3.7· (Huma



java.io.*

```
n)\r\nContent-Type: ·application/presentation; ·charset  
=UTF-8\r\nContent-Length: ·3000\r  
\n\r\n...
```

Server: ·JakeWharton/2.3.7·(Huma



Server: ·JakeWharton/2.3.7·(Huma



java.io.*

=UTF-8\r\nContent-Length: 3000\r\n\r\n...

Server: JakeWharton/2.3.7 (Human)rnContent-Type: application/presentation; charset



Server: JakeWharton/2.3.7 (Huma



java.io.*

=UTF-8\r\nContent-Length: 3000\r\n\r\n...

Server: JakeWharton/2.3.7 (Human)rnContent-Type: application/presentation; charset



Server: JakeWharton/2.3.7 (Huma



java.io.*

=UTF-8\r\nContent-Length: 3000\r\n\r\n...

Server: JakeWharton/2.3.7 (Human) \r\nContent-Type: application/presentation; charset



Server: JakeWharton/2.3.7 (Human) \r\nContent-Type: app



java.io.*

=UTF-8\r\nContent-Length: 3000\r\n\r\n...

Server: JakeWharton/2.3.7 (Human) \r\nContent-Type: application/presentation; charset



Server: JakeWharton/2.3.7 (Human) \r\nContent-Type: app



java.io.*

=UTF-8\r\nContent-Length: 3000\r\n\r\n...

Server: JakeWharton/2.3.7 (Human)\r\nContent-Type: application/presentation; charset



Content-Type: app



java.io.*

=UTF-8\r\nContent-Length: 3000\r\n\r\n...

Server: JakeWharton/2.3.7 (Human)\r\nContent-Type: application/presentation; charset



Content-Type: app



java.io.*

=UTF-8\r\nContent-Length: 3000\r\n\r\n...

Server: JakeWharton/2.3.7 (Human)\r\nContent-Type: application/presentation; charset



lication/presentation; charset



Content-Type: app



java.io.*

=UTF-8\r\nContent-Length: 3000\r\n\r\n...

Server: JakeWharton/2.3.7 (Human)\r\nContent-Type: application/presentation; charset



lication/presentation; charset Content-Type: app



java.io.*

- API is conflated with multiple concepts and inconsistent behavior
- Burden of figuring out storage is on the consumer

java.io.*

- API is conflated with multiple concepts and inconsistent behavior
- Burden of figuring out storage is on the consumer
- Decorators add performance, functionality

java.io.*

```
class DataInputStream extends InputStream {
    DataInputStream(InputStream in) {}

    void readFully(byte[] b) {}
    void readFully(byte b[], int off, int len) {}
    boolean readBoolean() {}
    byte readByte() {}
    int readUnsignedByte() {}
    short readShort() {}
    int readUnsignedShort() {}
    char readChar() {}
    int readInt() {}
    long readLong() {}
    float readFloat() {}
    double readDouble() {}
    String readLine() {}
    String readUTF() {}
}
```

java.io.*

```
class DataInputStream extends InputStream {
    DataInputStream(InputStream in) {}

    void readFully(byte[] b) {}
    void readFully(byte b[], int off, int len) {}
    boolean readBoolean() {}
    byte readByte() {}
    int readUnsignedByte() {}
    short readShort() {}
    int readUnsignedShort() {}
    char readChar() {}
    int readInt() {}
    long readLong() {}
    float readFloat() {}
    double readDouble() {}
    String readLine() {}
    String readUTF() {}
}
```

```
class InputStreamReader extends Reader {
    InputStreamReader(InputStream is) {}
    InputStreamReader(InputStream is, String charsetName) {}
    InputStreamReader(InputStream is, CharsetDecoder dec) {}
    InputStreamReader(InputStream is, Charset charset) {}

    int read() {}
    int read(char[] buffer) {}
    int read(char[] buffer, int offset, int count) {}
    int read(CharBuffer buffer) {}
}
```

java.io.*

- API is conflated with multiple concepts and inconsistent behavior
- Burden of figuring out storage is on the consumer
- Decorators add performance, functionality

java.io.*

- API is conflated with multiple concepts and inconsistent behavior
- Burden of figuring out storage is on the consumer
- Decorators add performance, functionality
- High-level abstractions exist, but do not compose

Okio

Okio

- Complements `java.io.*` and `java.nio.*`

Okio

- Complements `java.io.*` and `java.nio.*`
- `Source` and `Sink` move data

Okio

```
interface Source {  
}
```

Okio

```
interface Source {  
    void close();  
}
```

Okio

```
interface Source {  
    Timeout timeout();  
    void close();  
}
```

Okio

```
interface Source {  
    long read(Buffer sink, long byteCount);  
    Timeout timeout();  
    void close();  
}
```

Okio

```
interface Source {  
    long read(Buffer sink, long byteCount);  
    Timeout timeout();  
    void close();  
}
```

Okio

```
interface Sink {  
}
```

Okio

```
interface Sink {  
    void write(Buffer sink, long byteCount);  
    Timeout timeout();  
    void close();  
}
```

Okio

```
interface Sink {  
    void write(Buffer sink, long byteCount);  
    Timeout timeout();  
    void close();  
    void flush();  
}
```


Okio

```
interface Sink {  
    void write(Buffer sink, long byteCount);  
    Timeout timeout();  
    void close();  
    void flush();  
}
```

Okio

- Complements `java.io.*` and `java.nio.*`
- `Source` and `Sink` move data

Okio

- Complements `java.io.*` and `java.nio.*`
- `Source` and `Sink` move data
- `Buffer` and `ByteString` hold data

Okio

```
Buffer buffer = new Buffer();
```

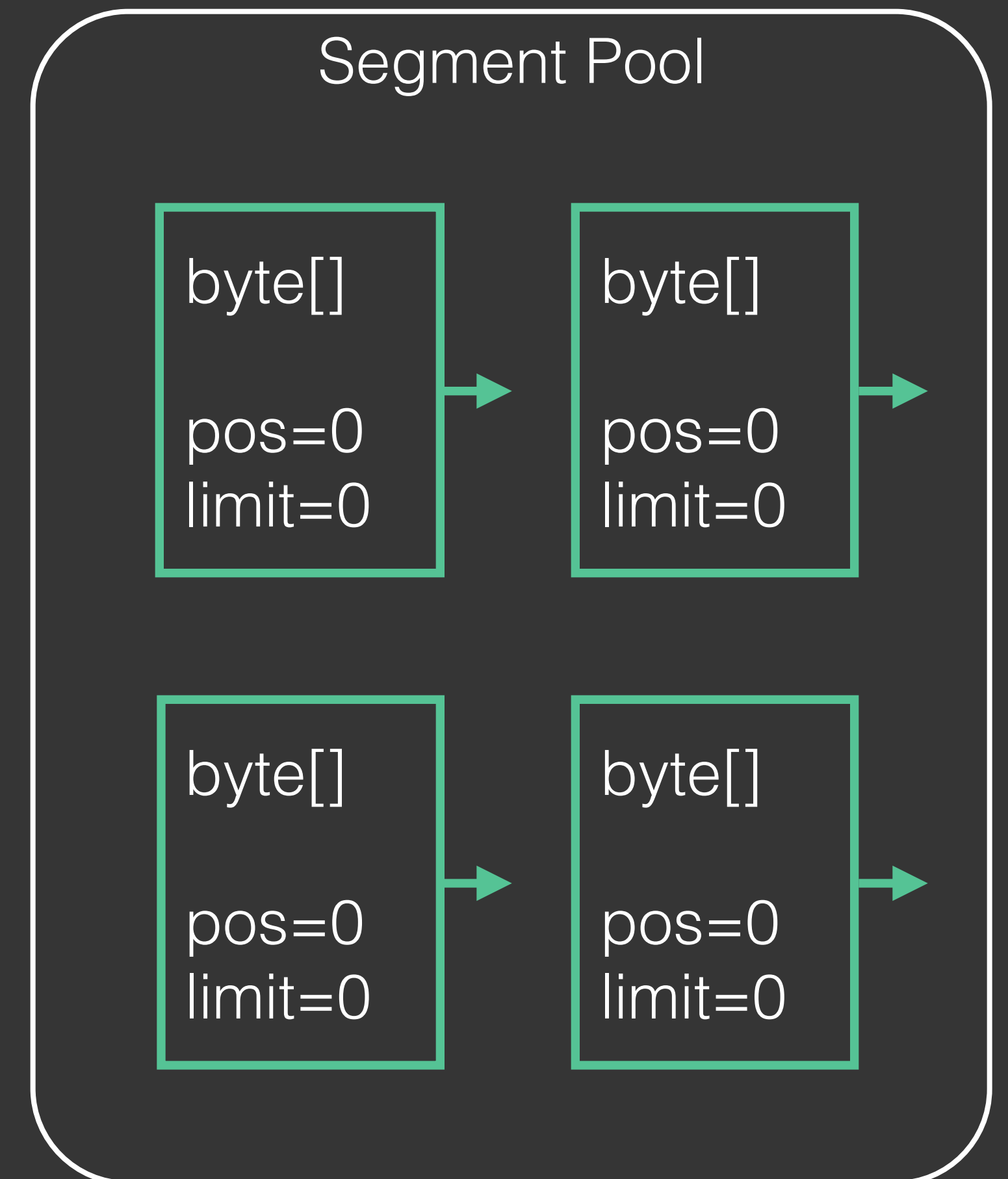
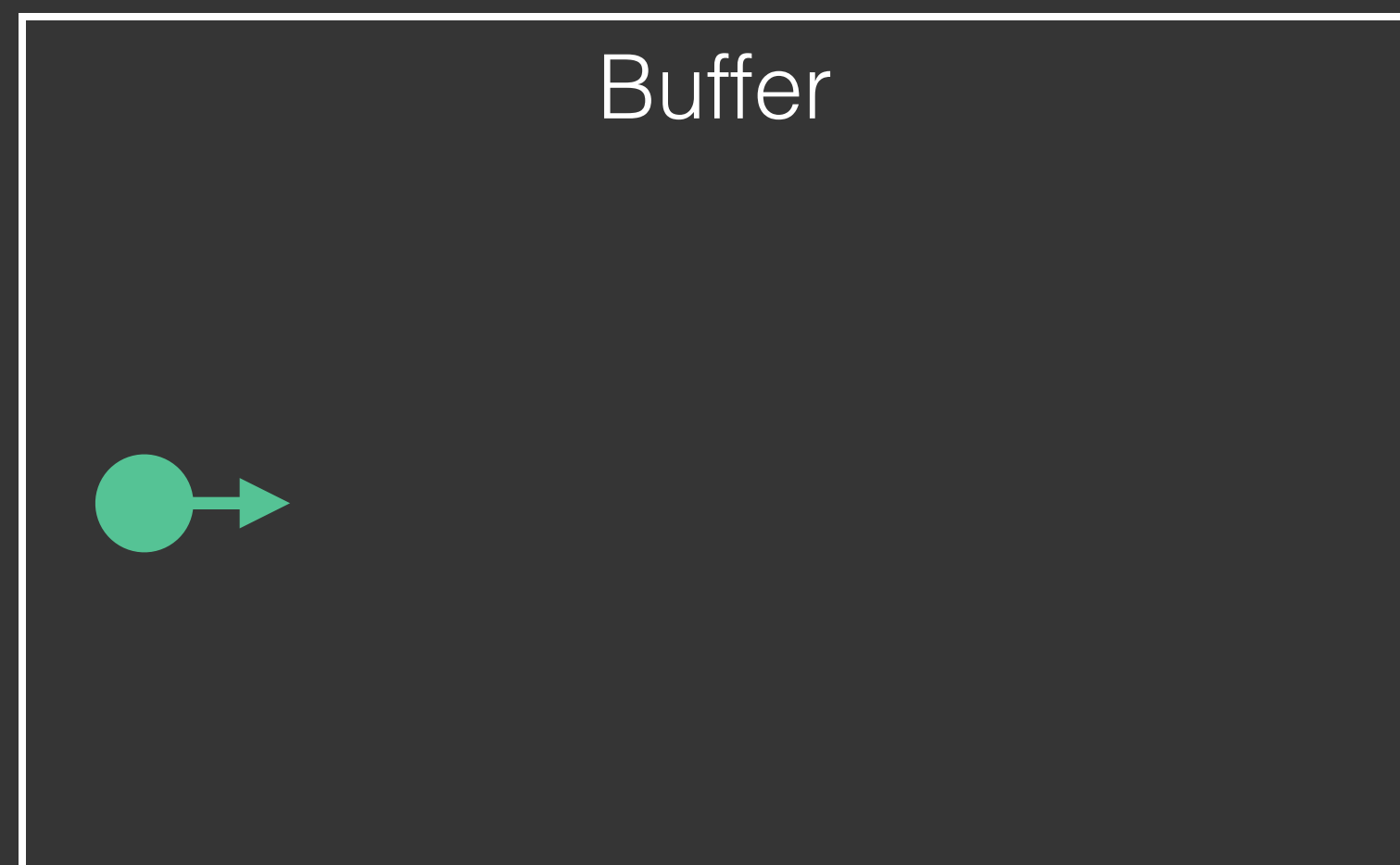
Okio

```
Buffer buffer = new Buffer();
```



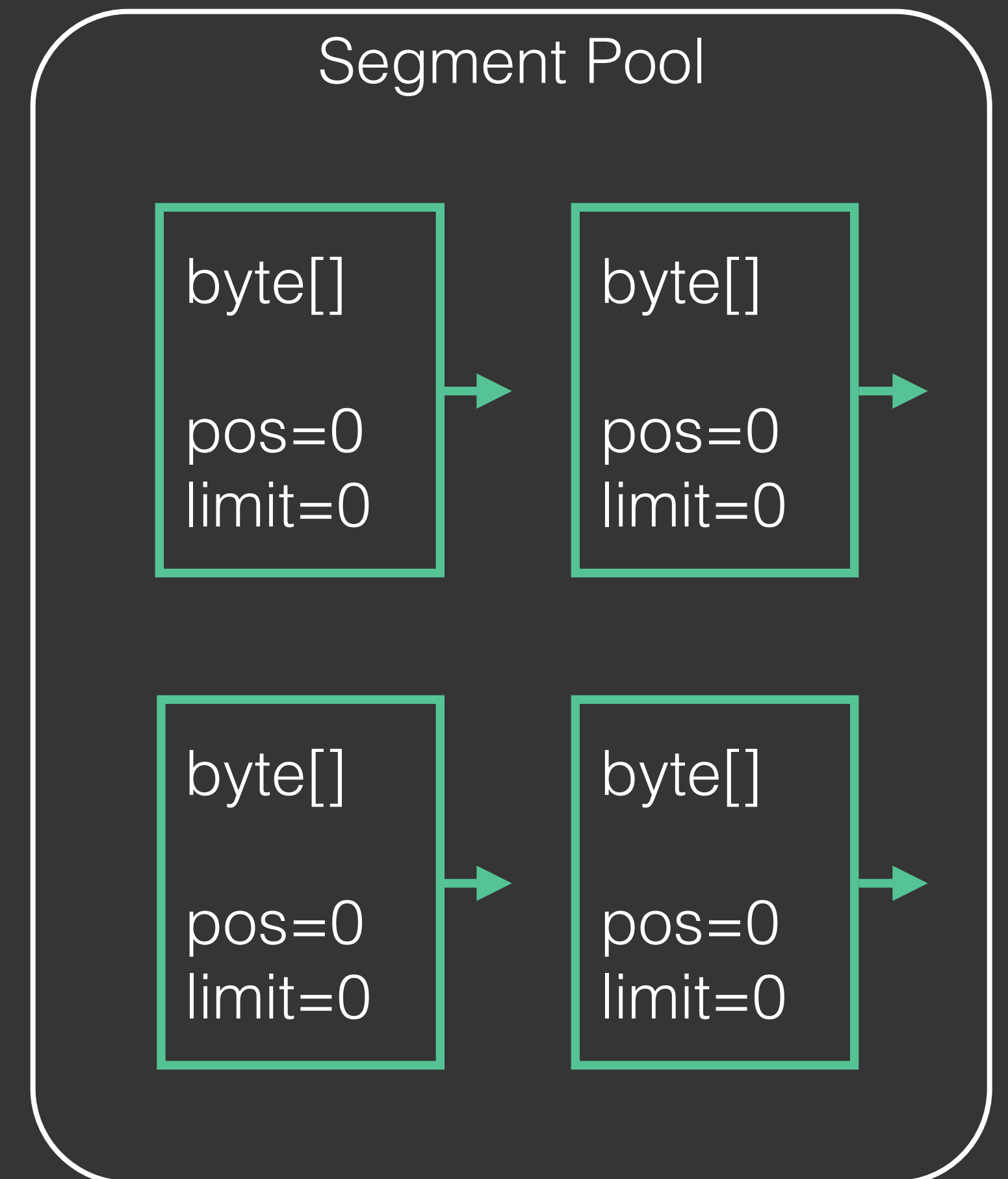
Okio

```
Buffer buffer = new Buffer();
```



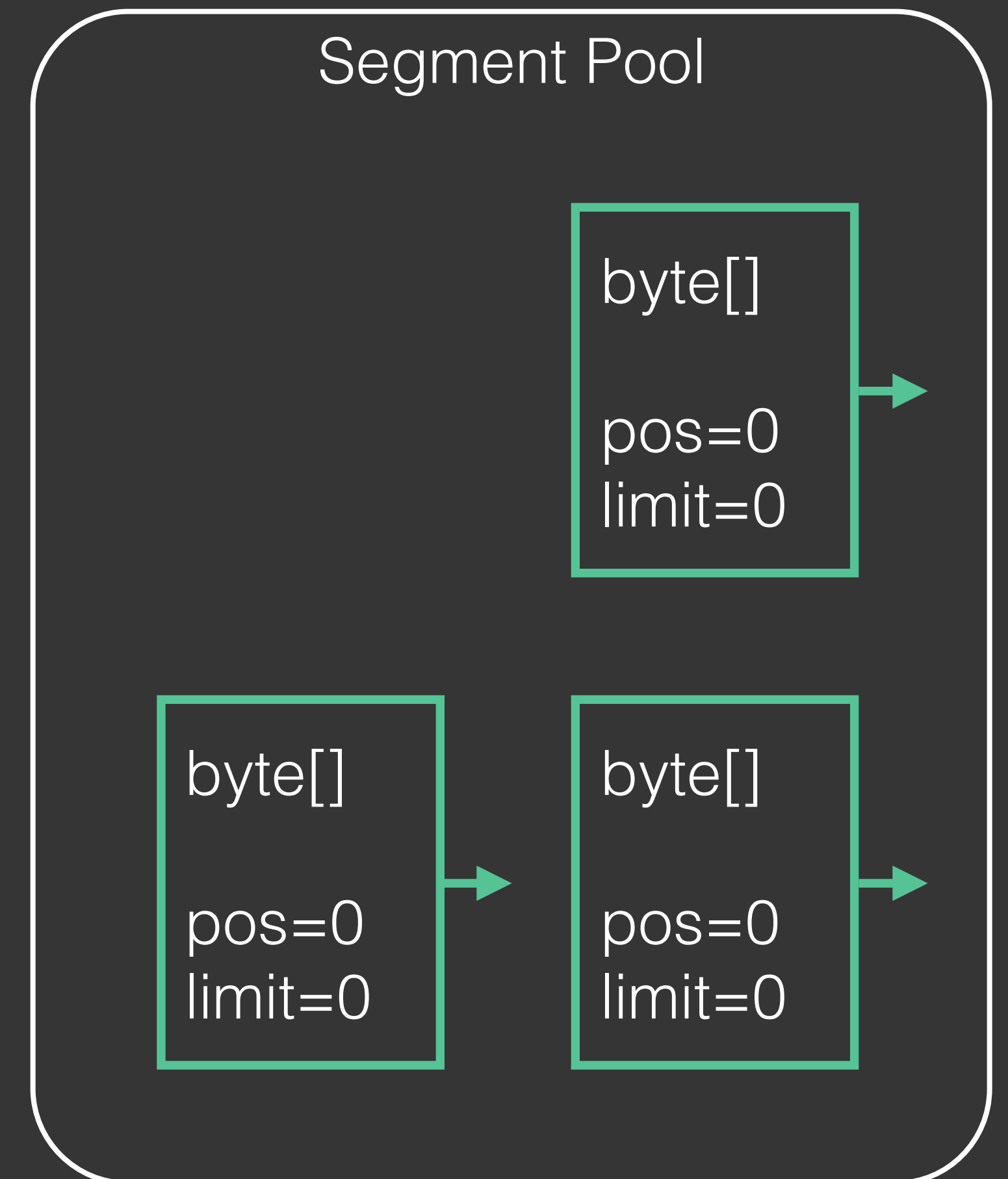
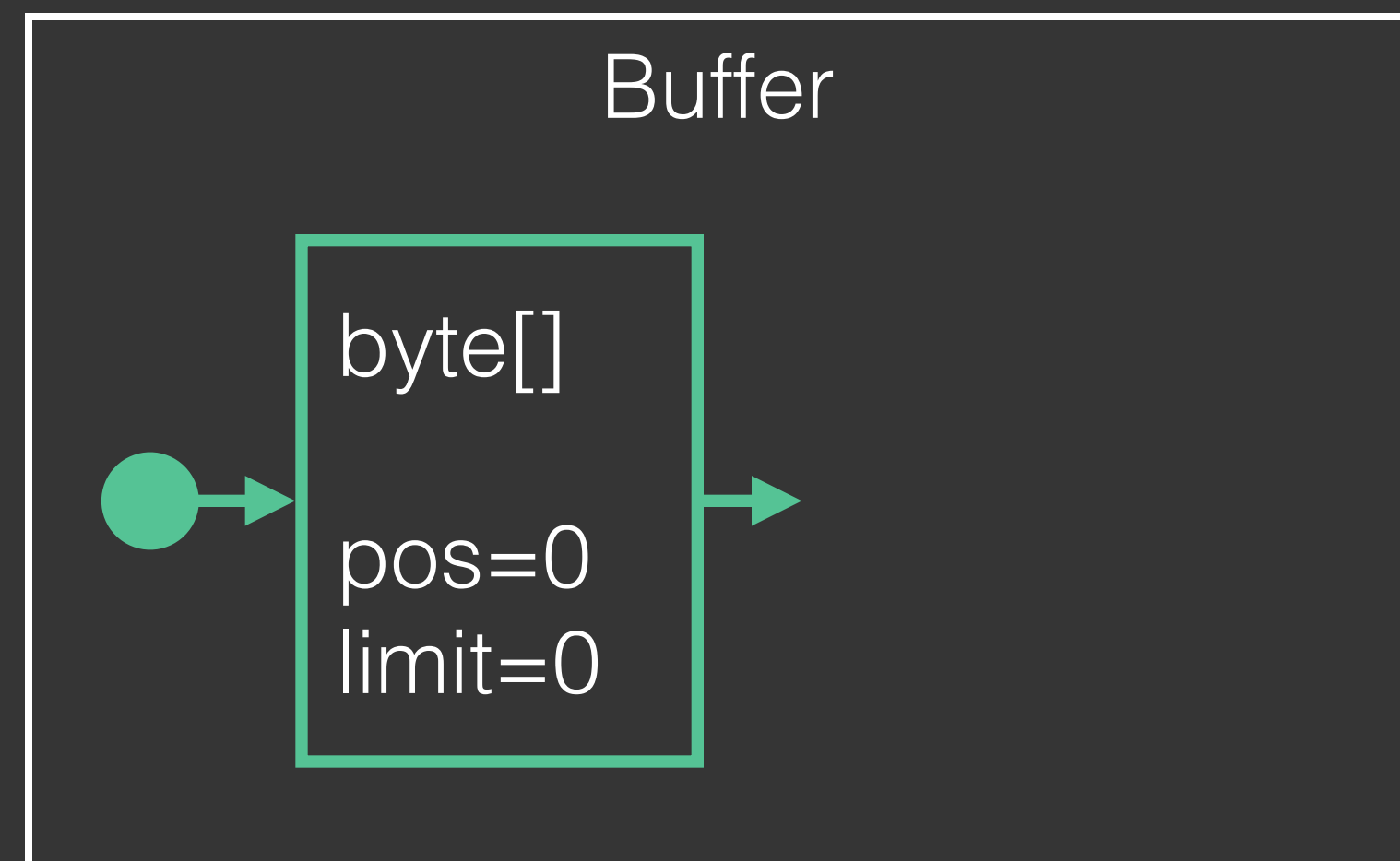
Okio

```
Buffer buffer = new Buffer();  
buffer.writeUtf8("Hello, Droidcon");
```



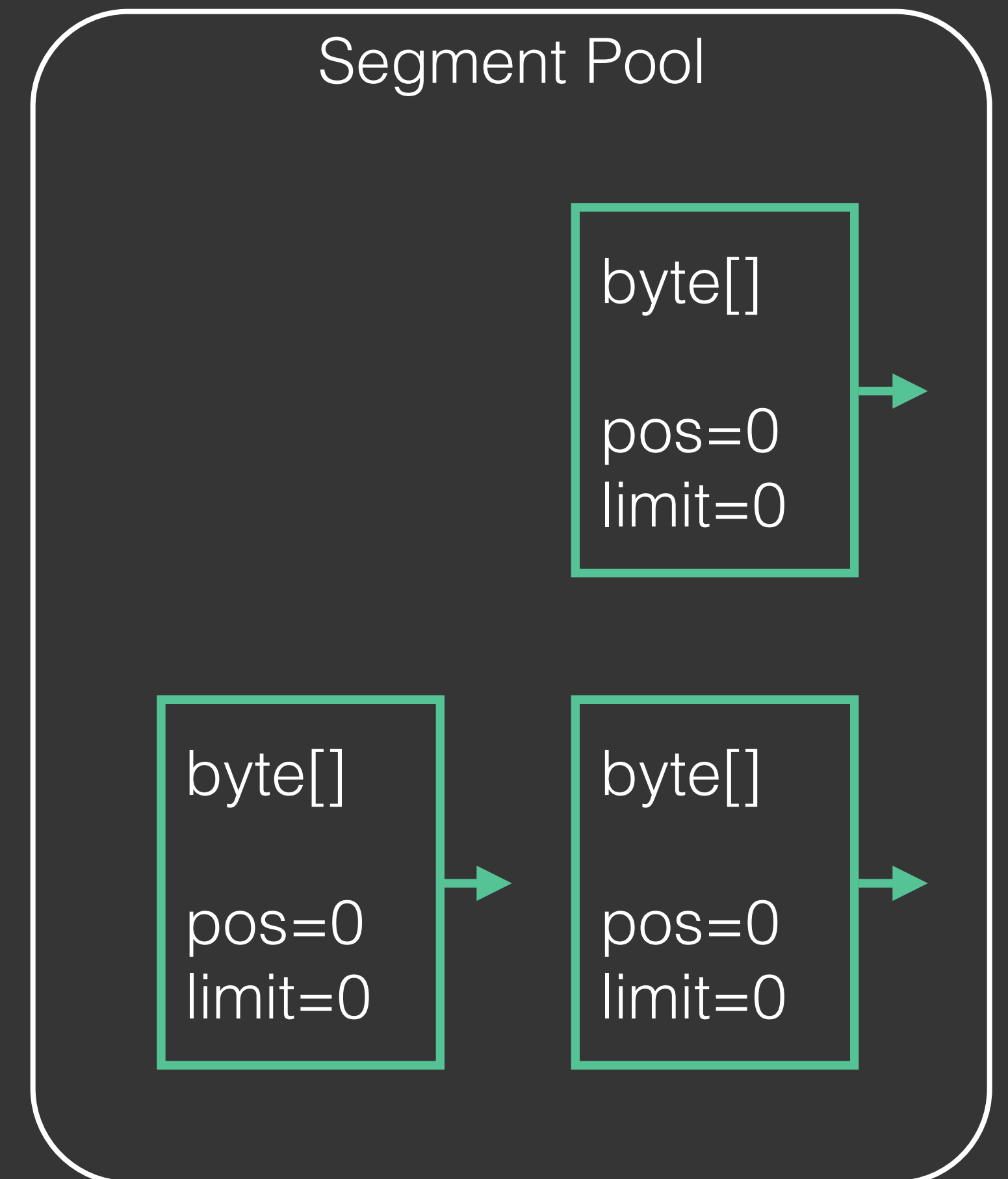
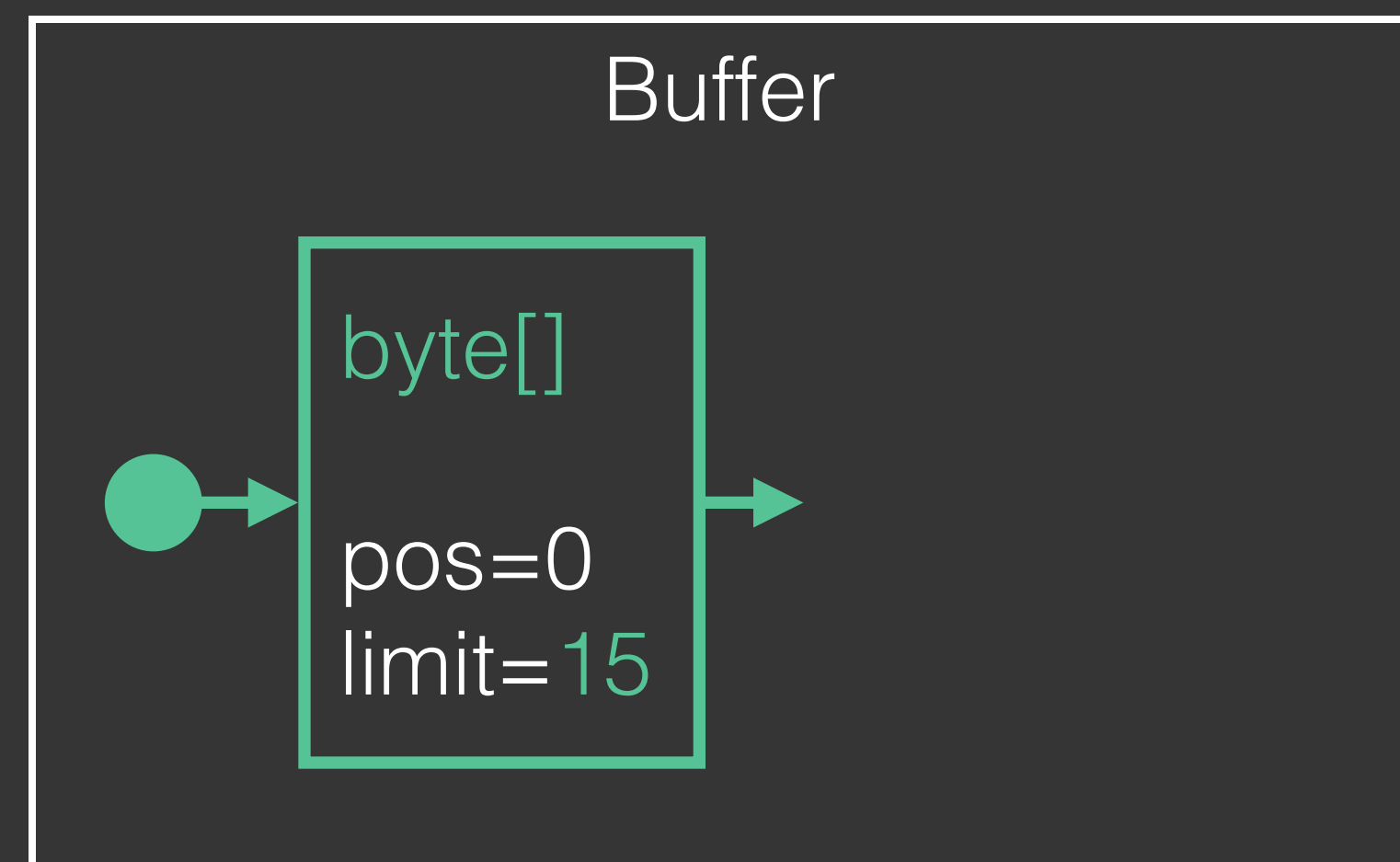
Okio

```
Buffer buffer = new Buffer();  
buffer.writeUtf8("Hello, Droidcon");
```



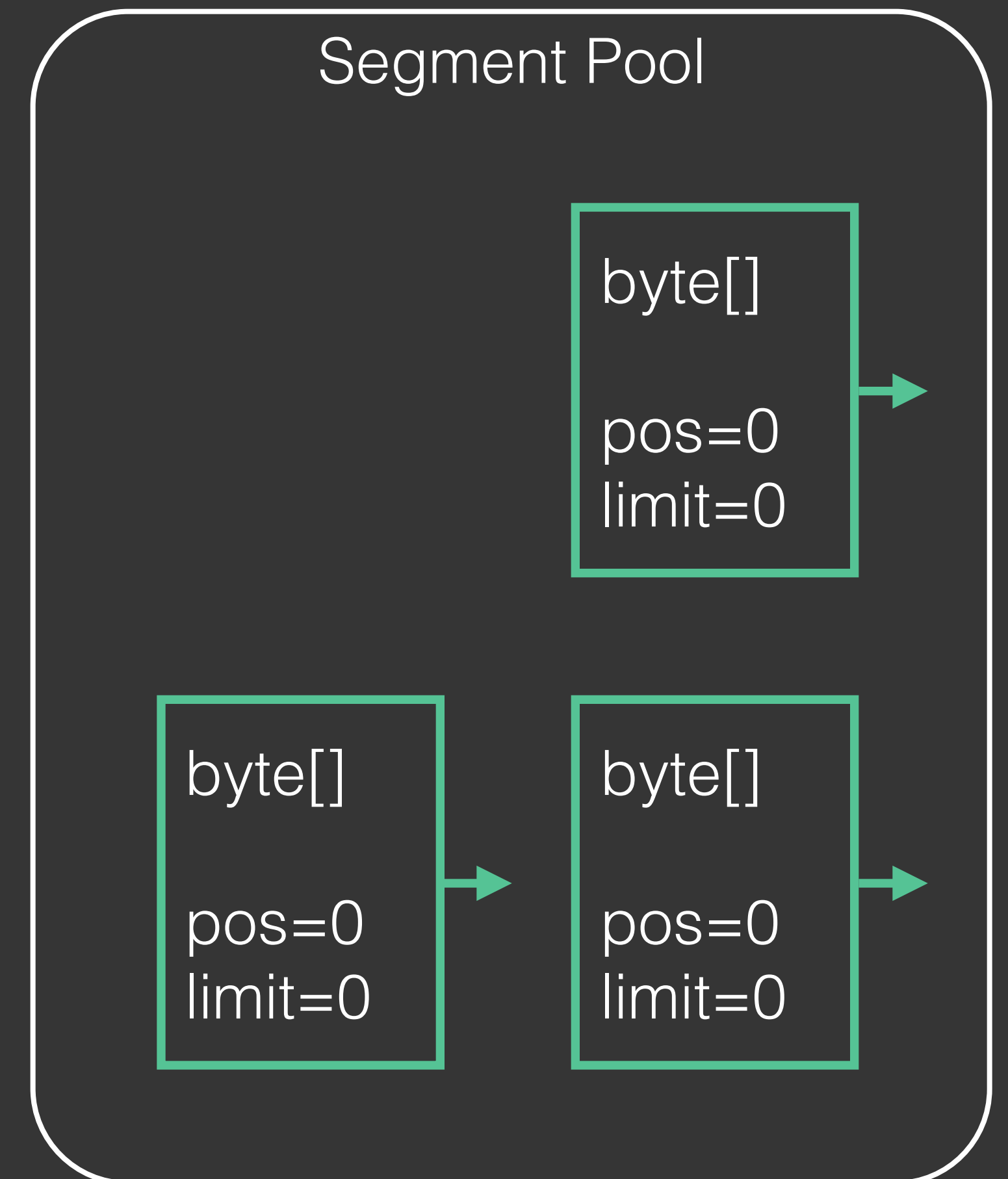
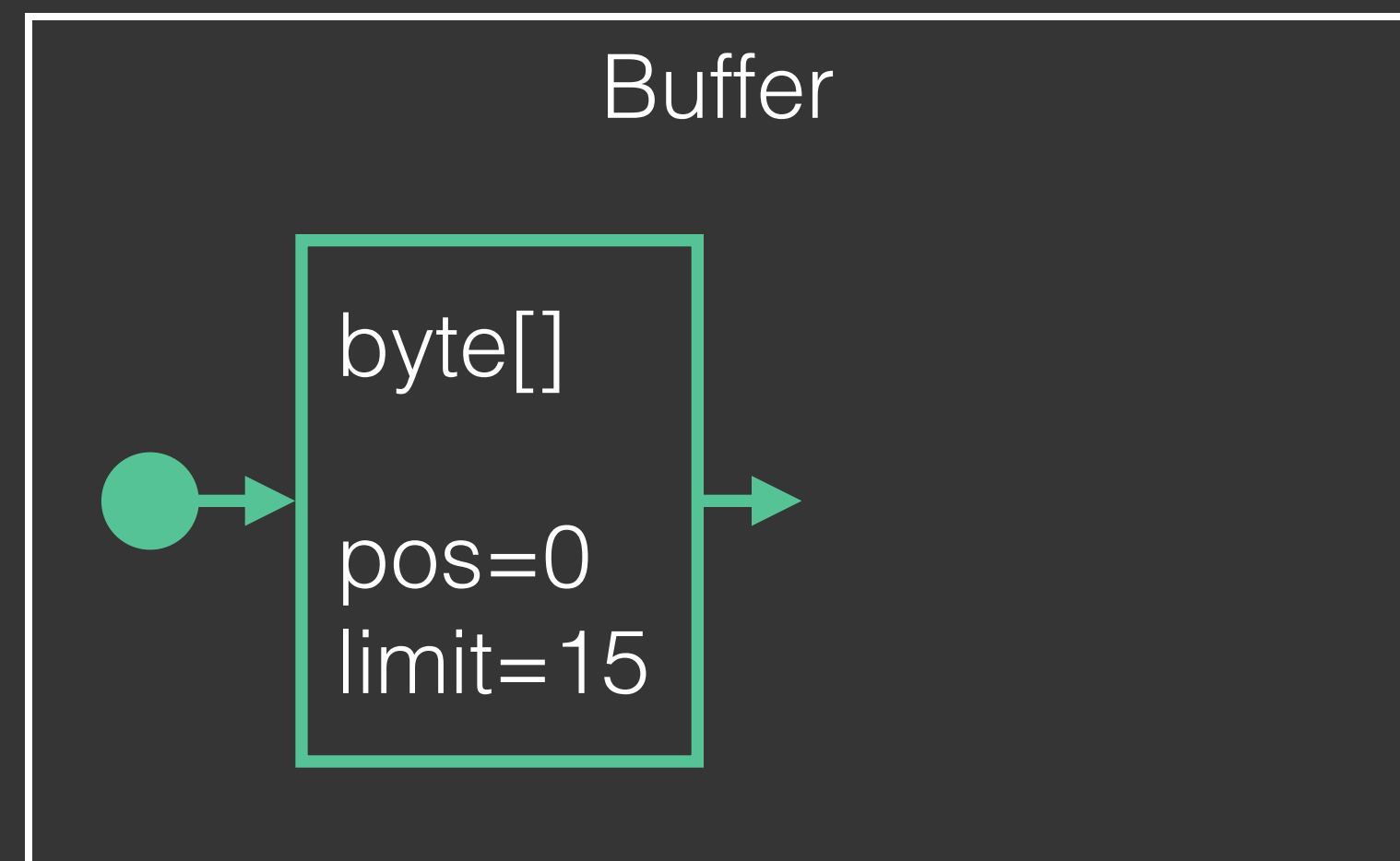
Okio

```
Buffer buffer = new Buffer();  
buffer.writeUtf8("Hello, Droidcon");
```



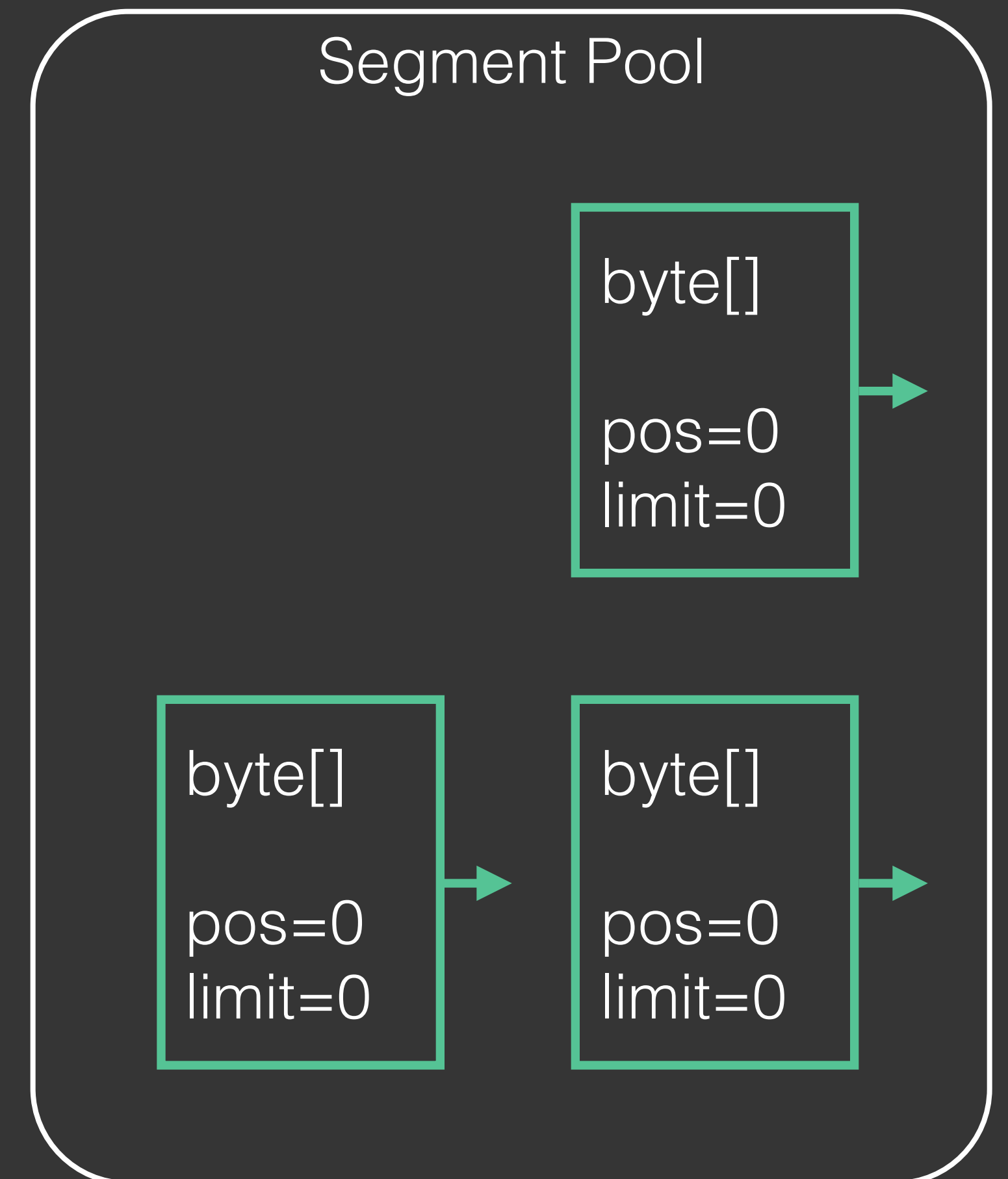
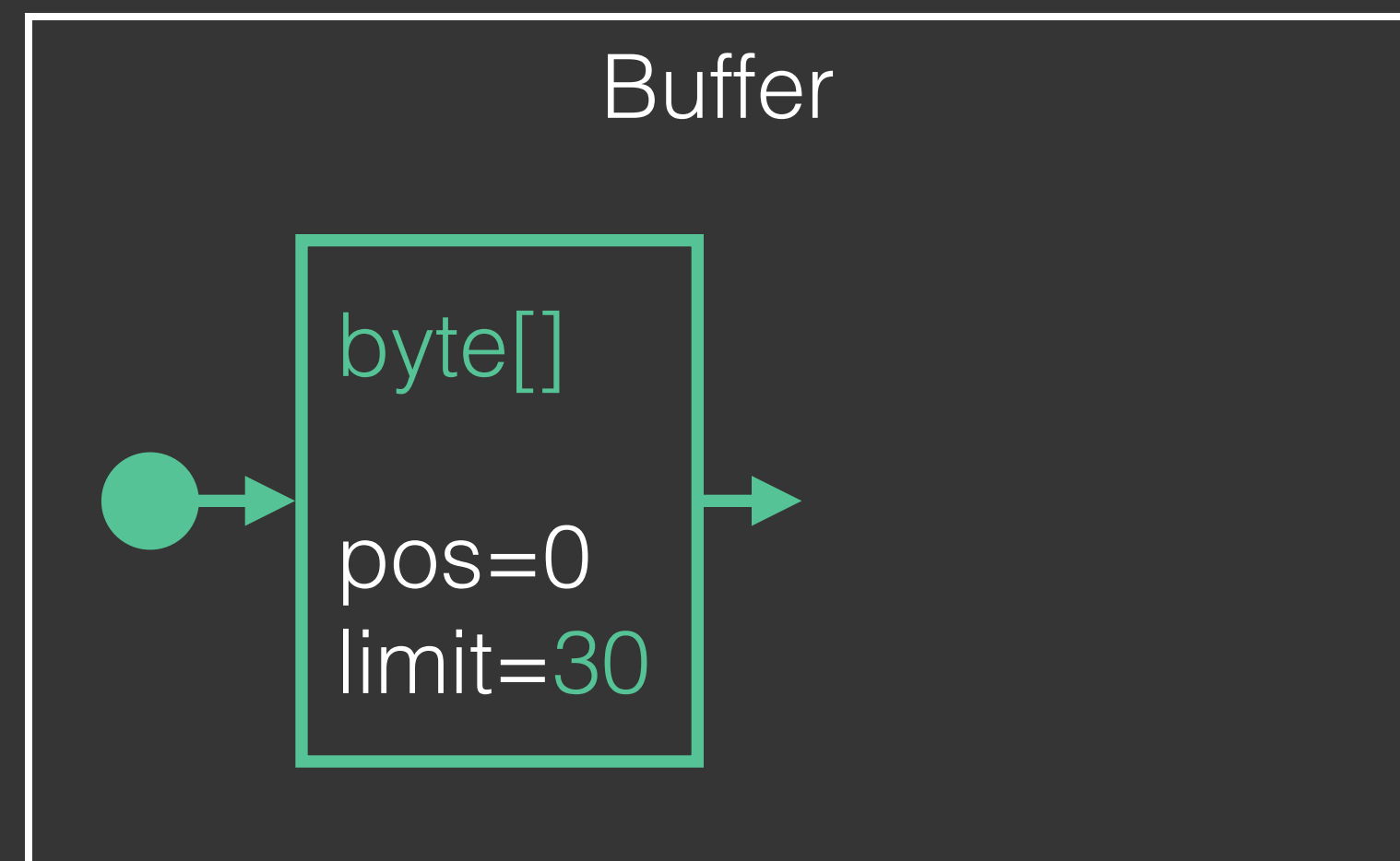
Okio

```
Buffer buffer = new Buffer();  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");
```



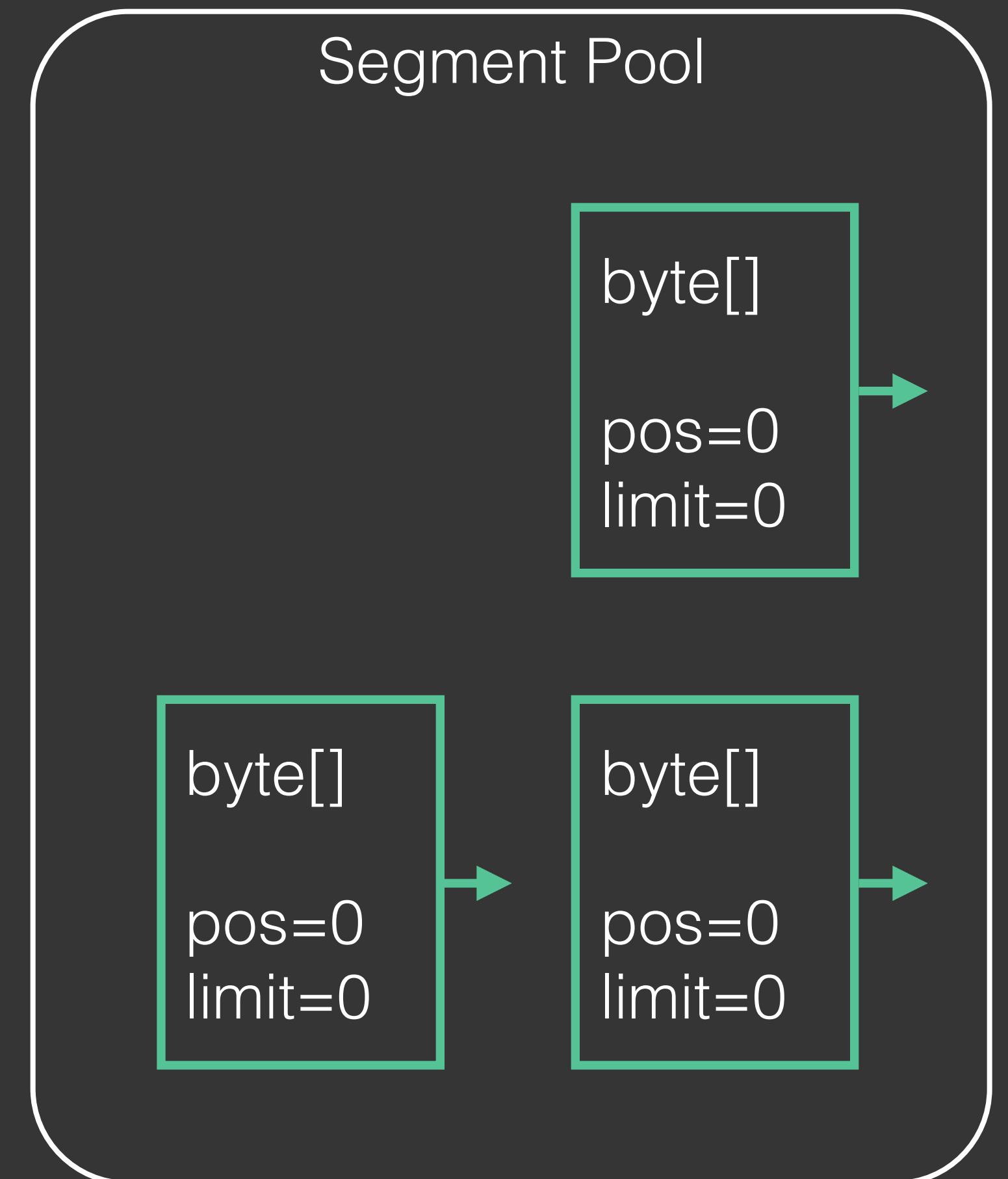
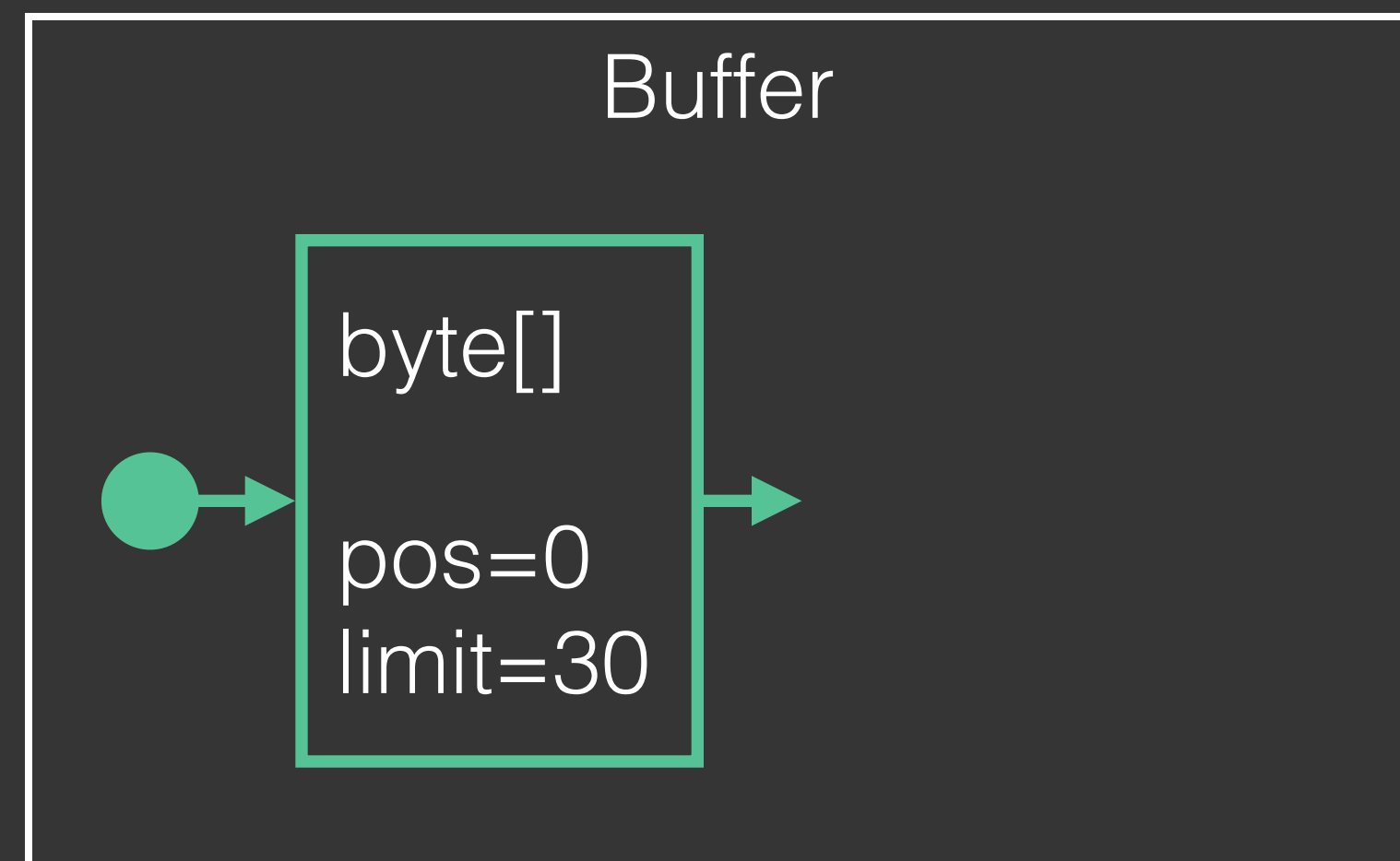
Okio

```
Buffer buffer = new Buffer();  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");
```



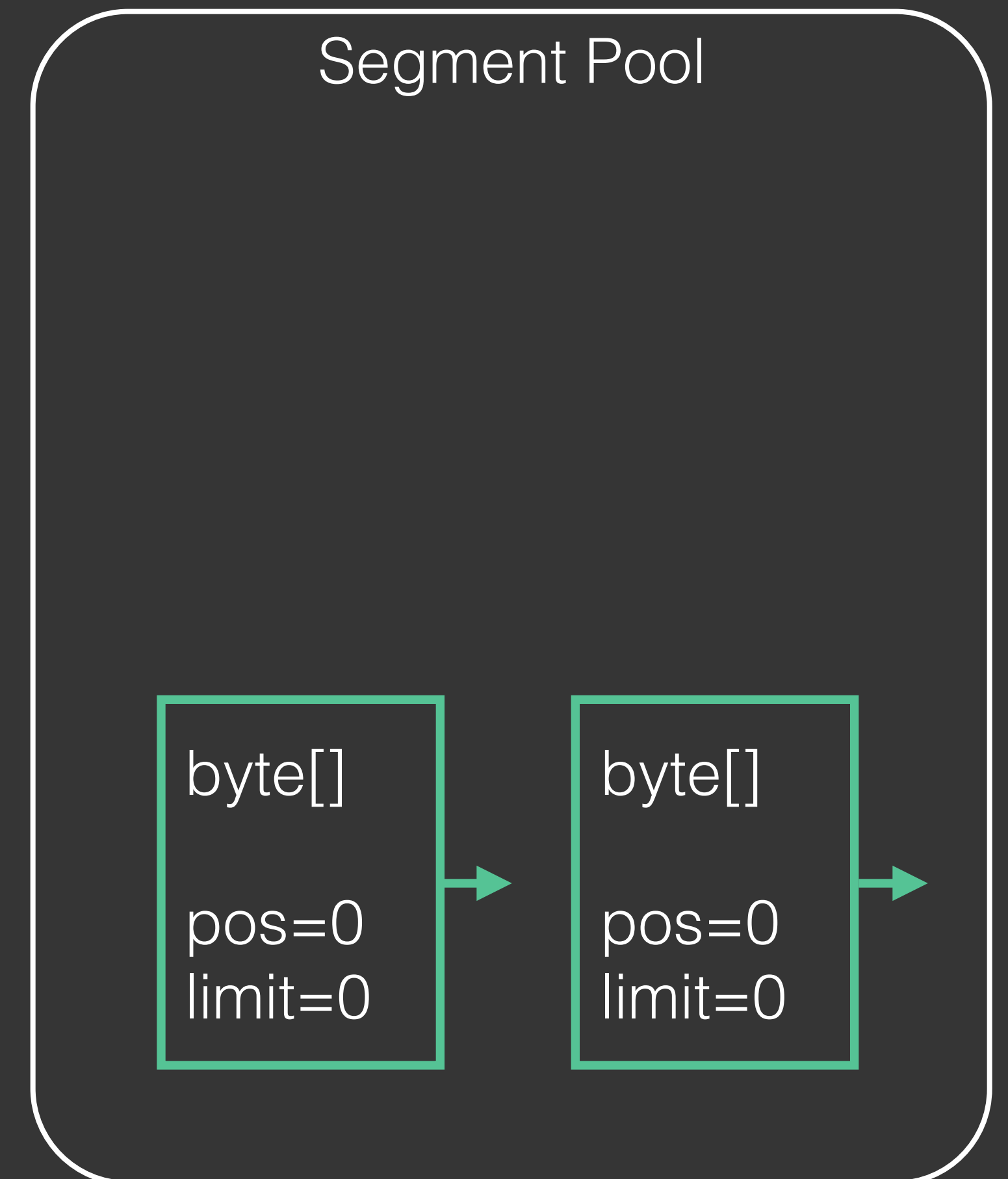
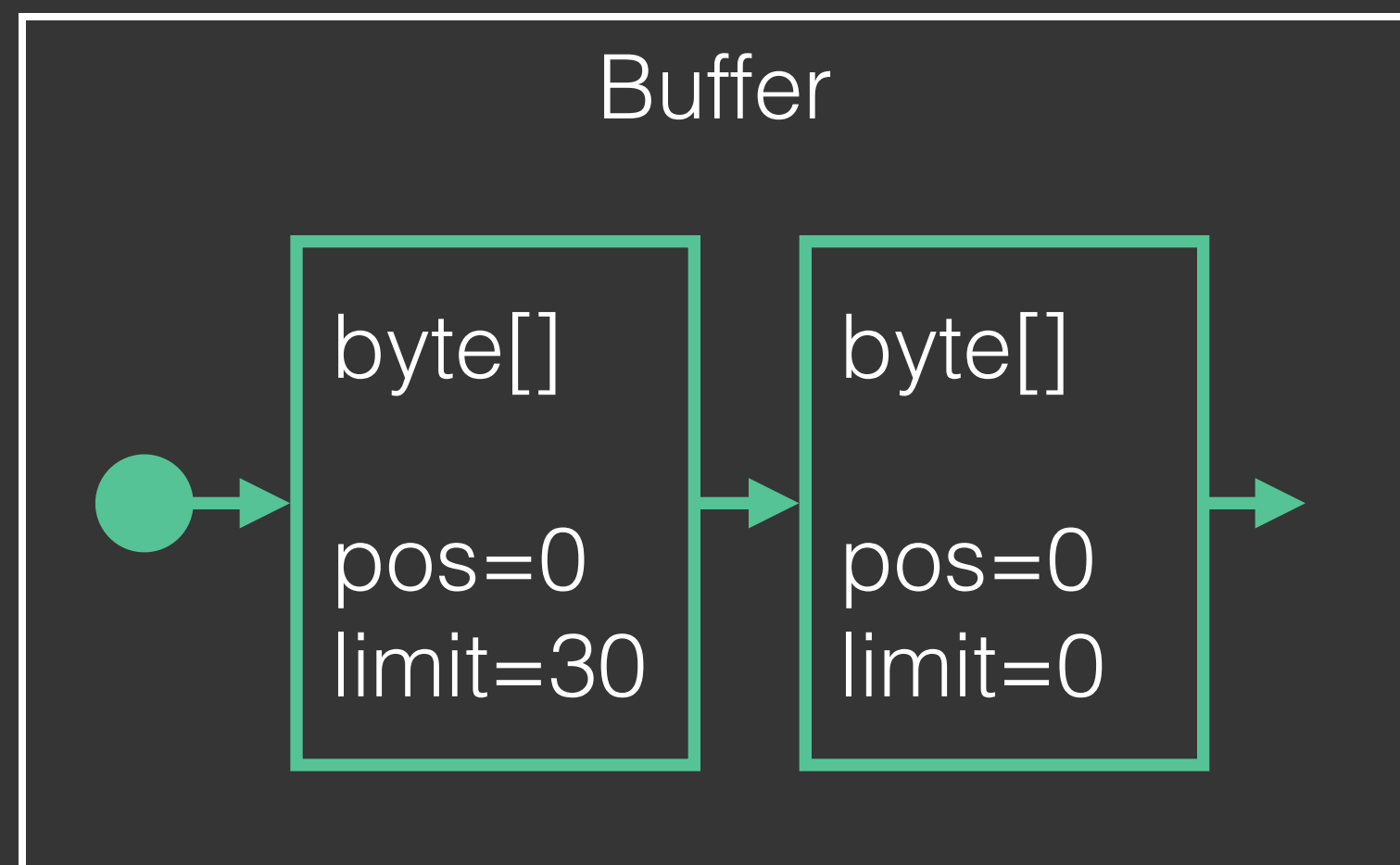
Okio

```
Buffer buffer = new Buffer();  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");
```



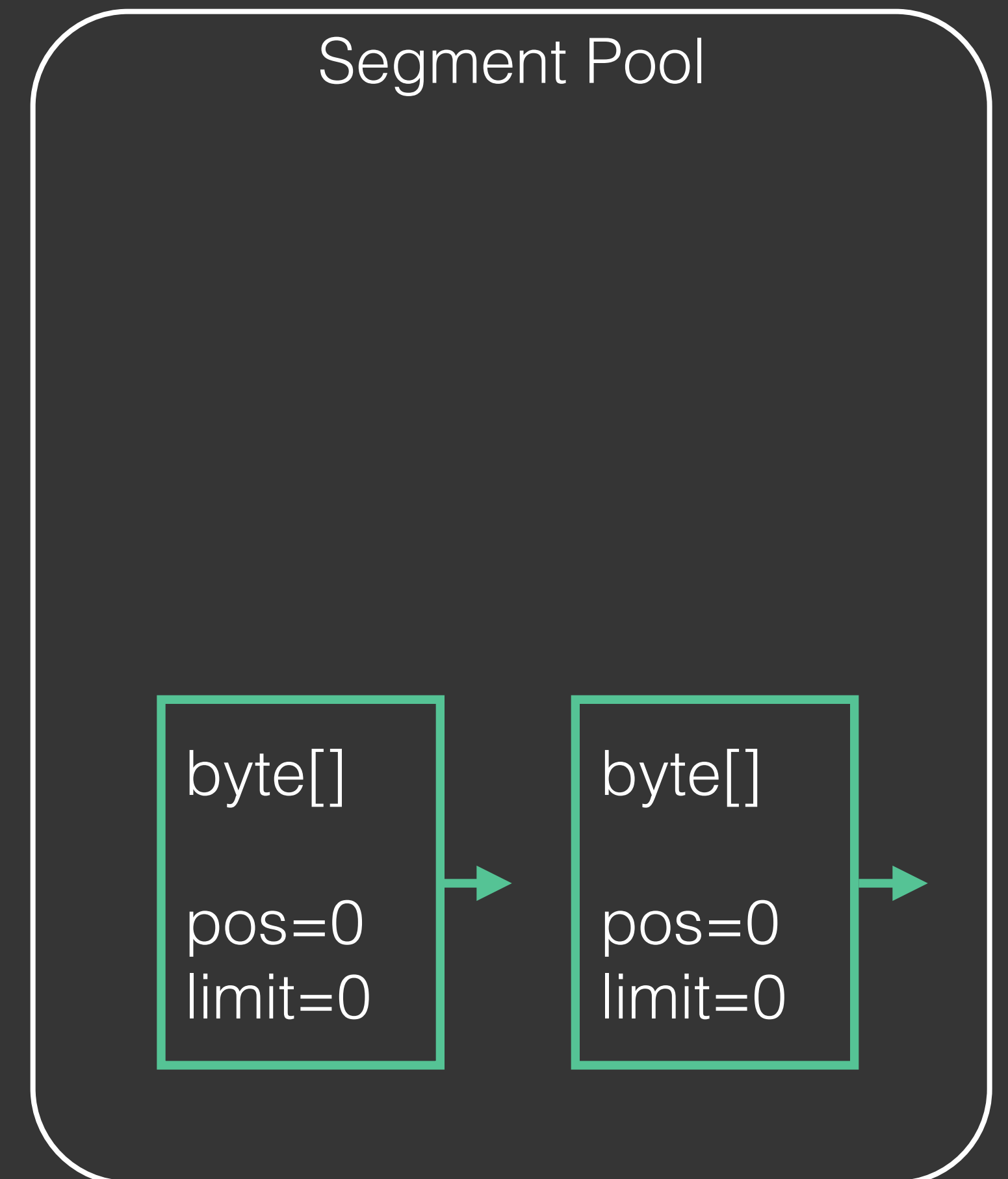
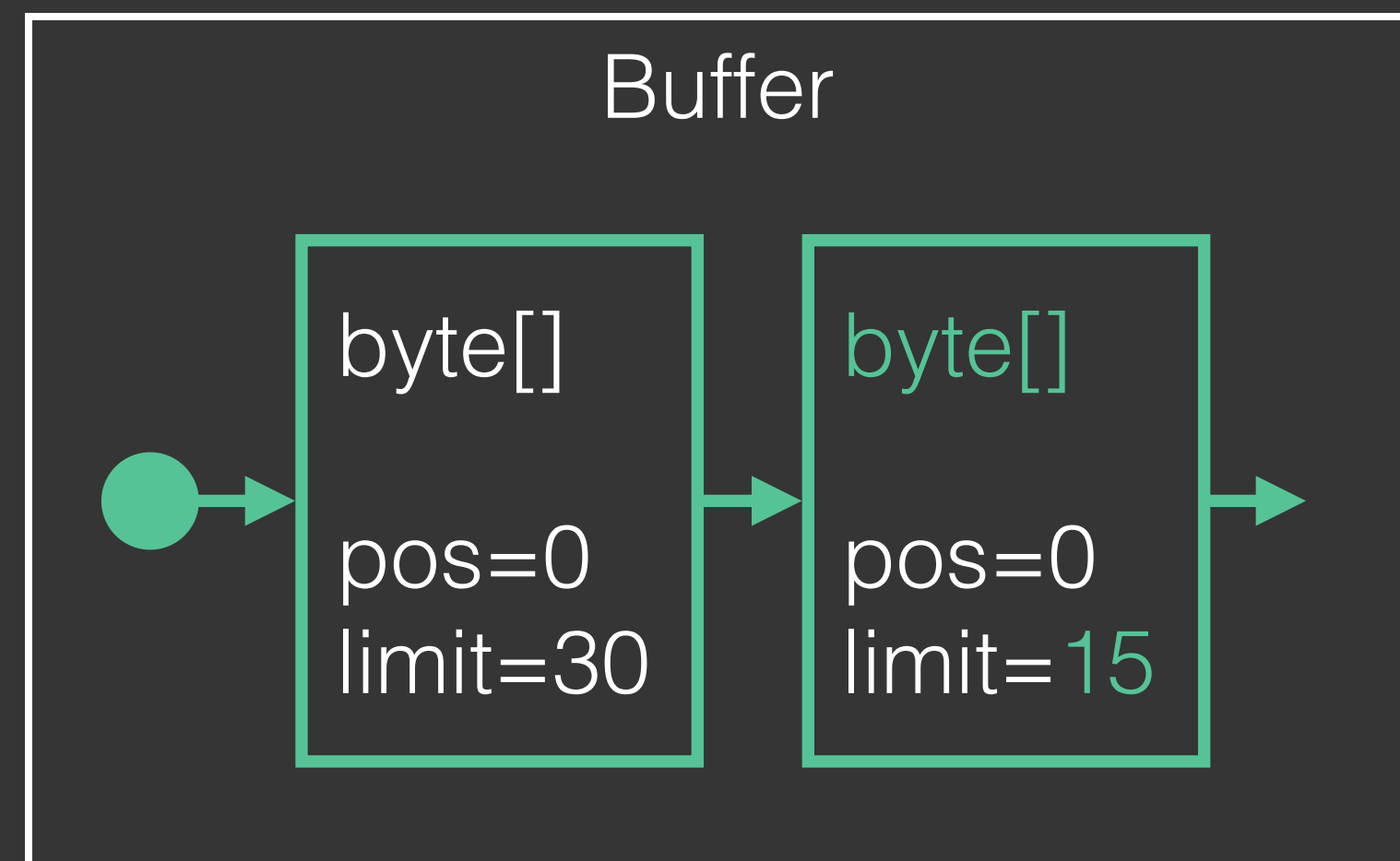
Okio

```
Buffer buffer = new Buffer();  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");
```



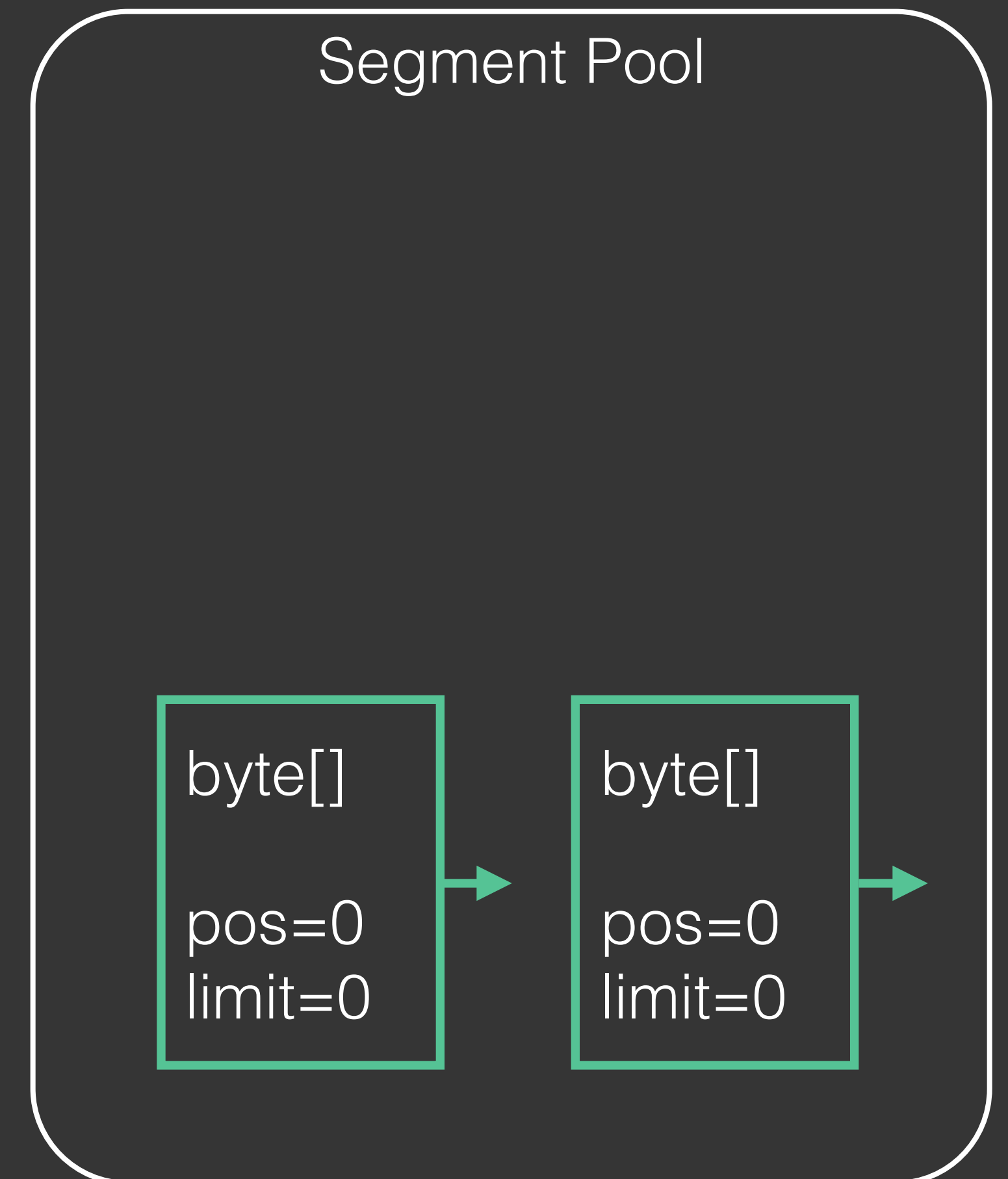
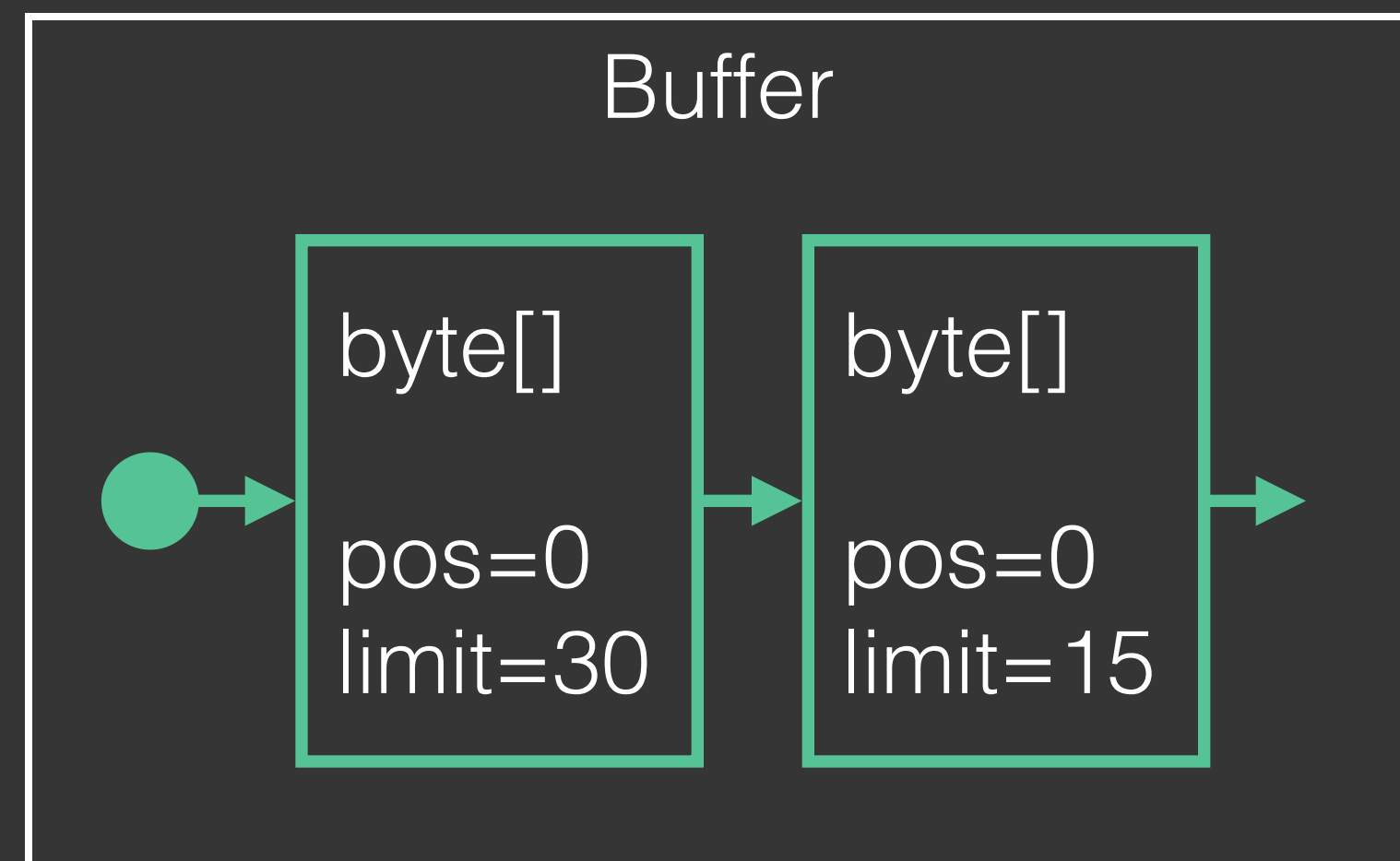
Okio

```
Buffer buffer = new Buffer();  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");
```



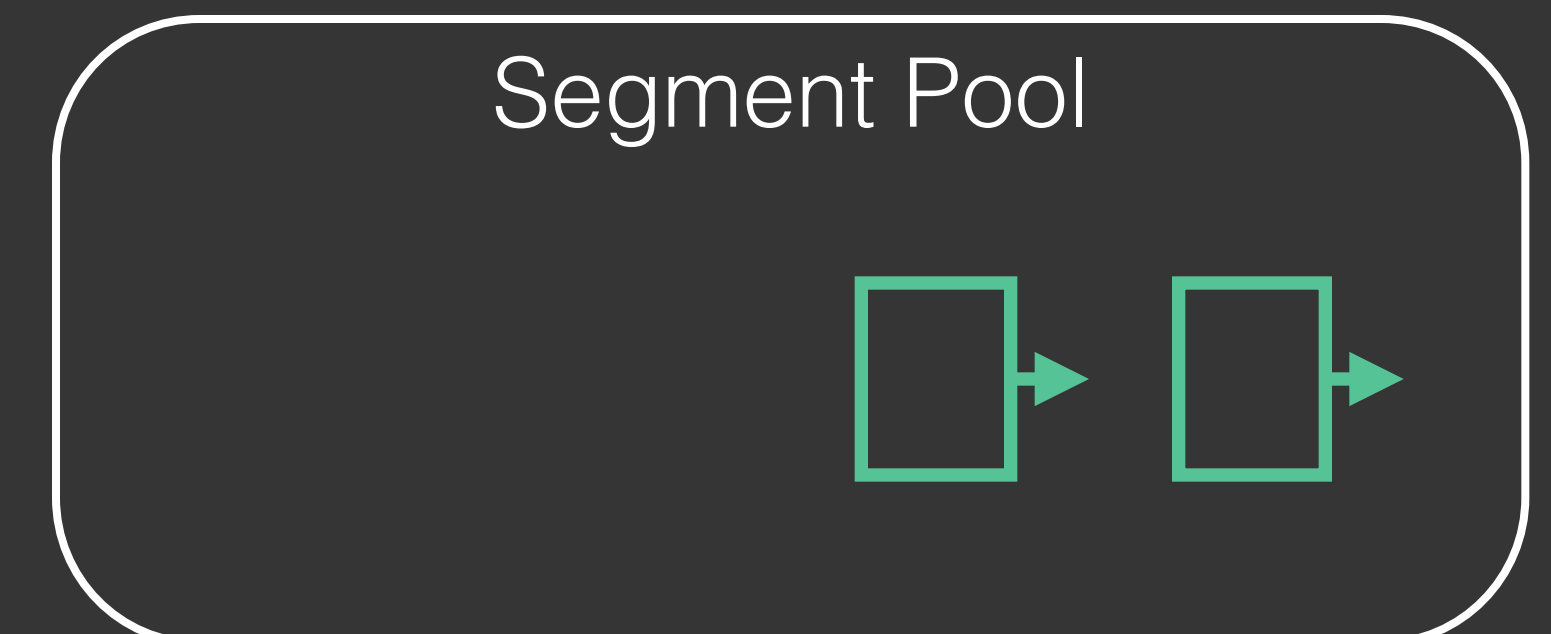
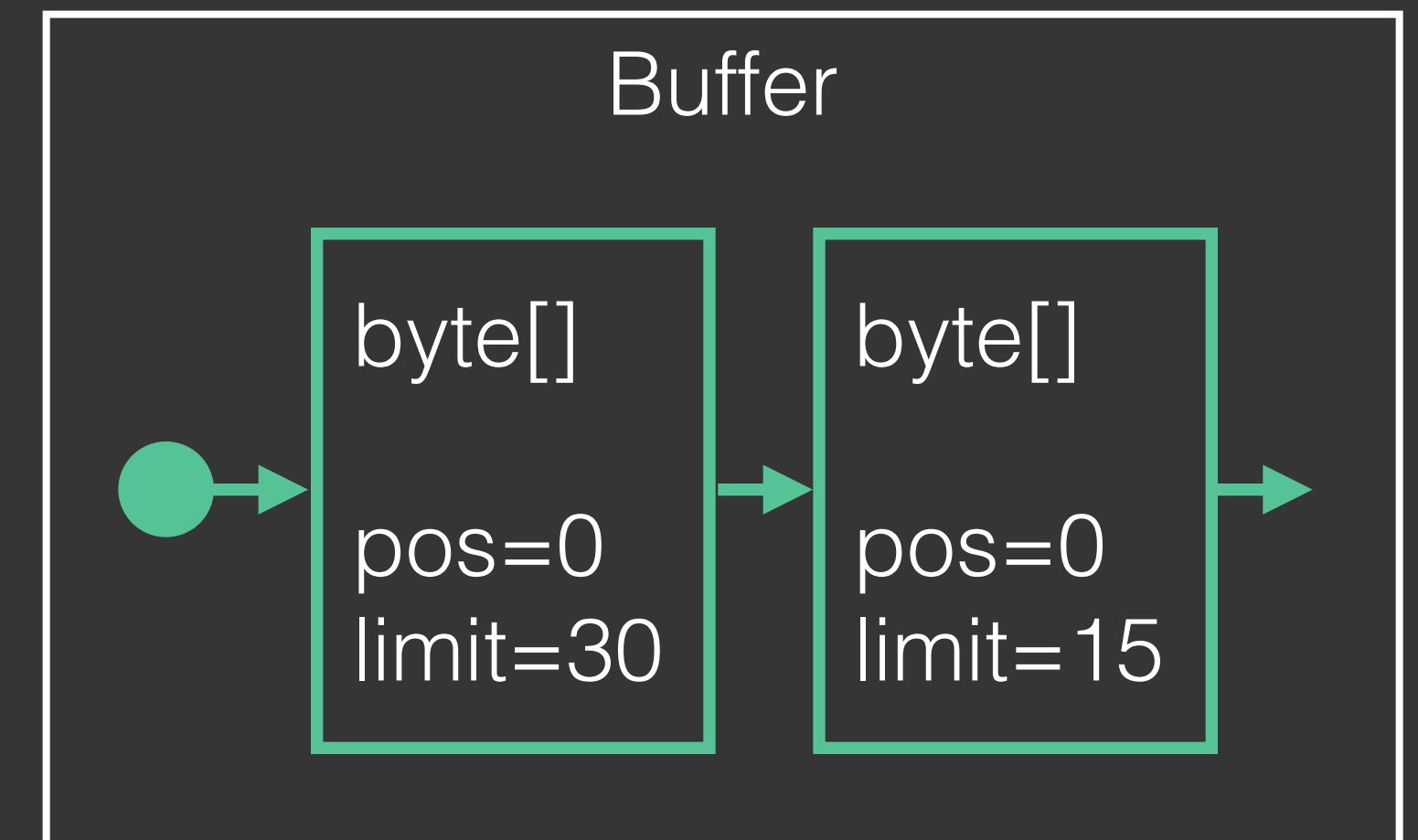
Okio

```
Buffer buffer = new Buffer();  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");
```



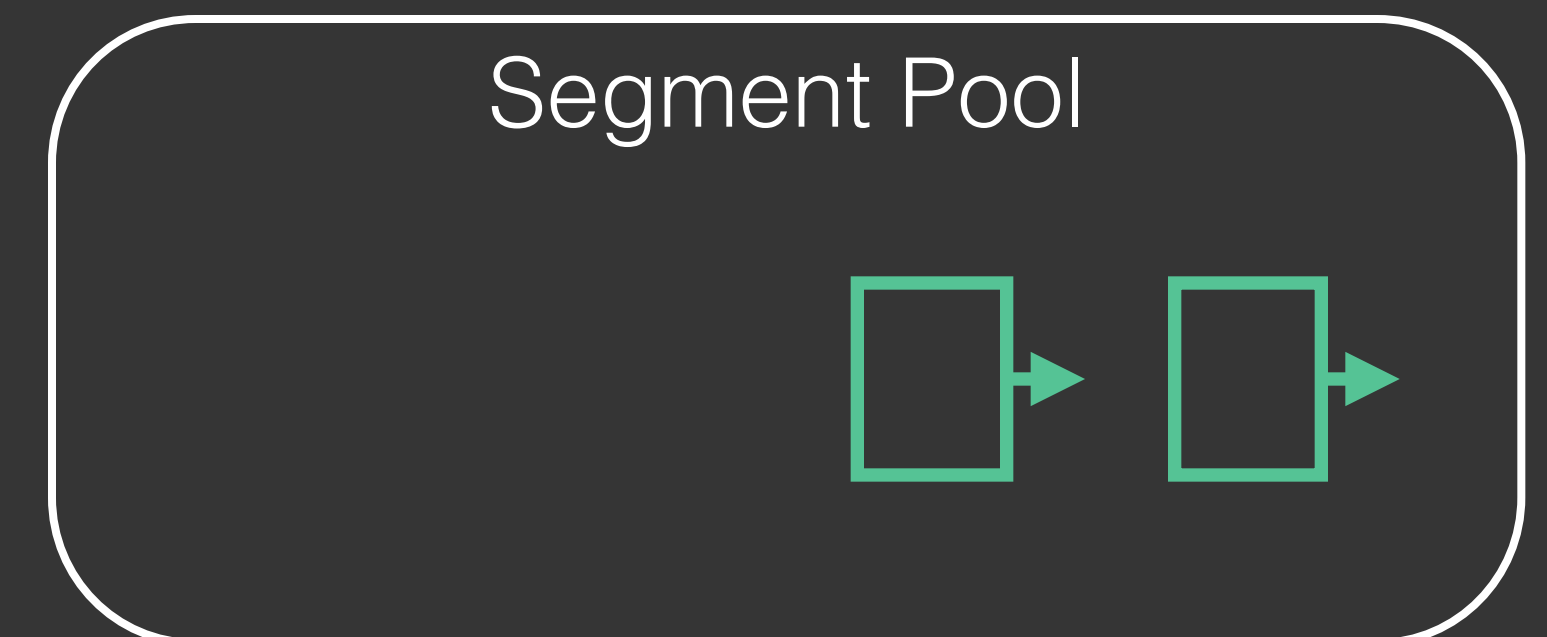
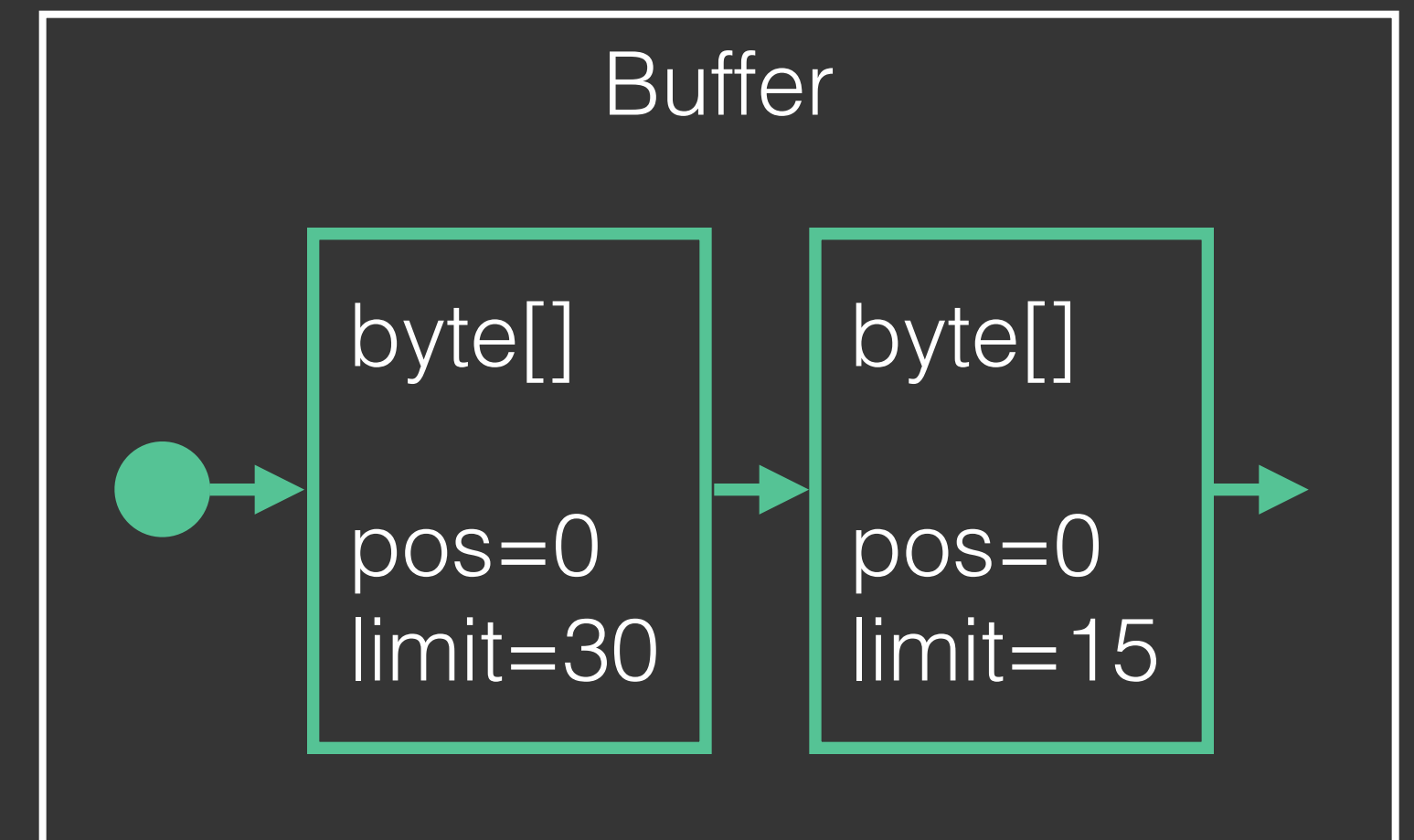
Okio

```
Buffer buffer = new Buffer();  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");
```



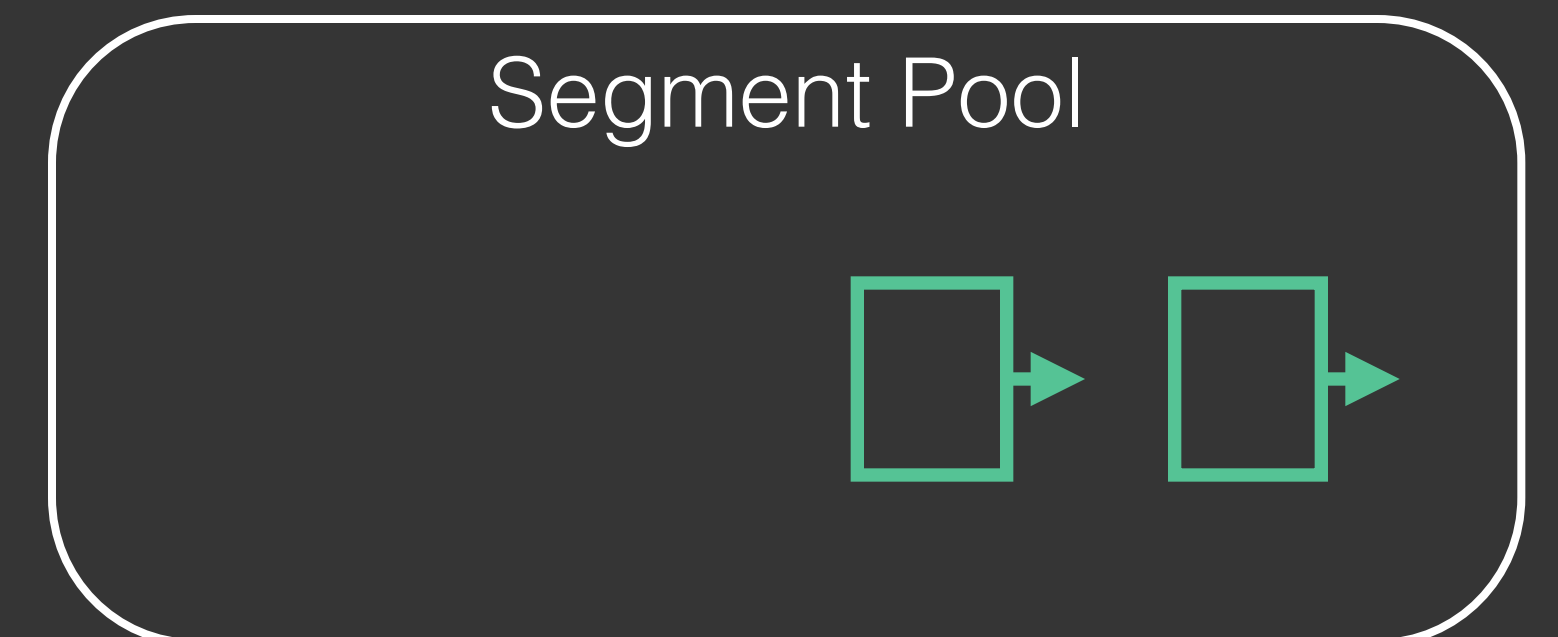
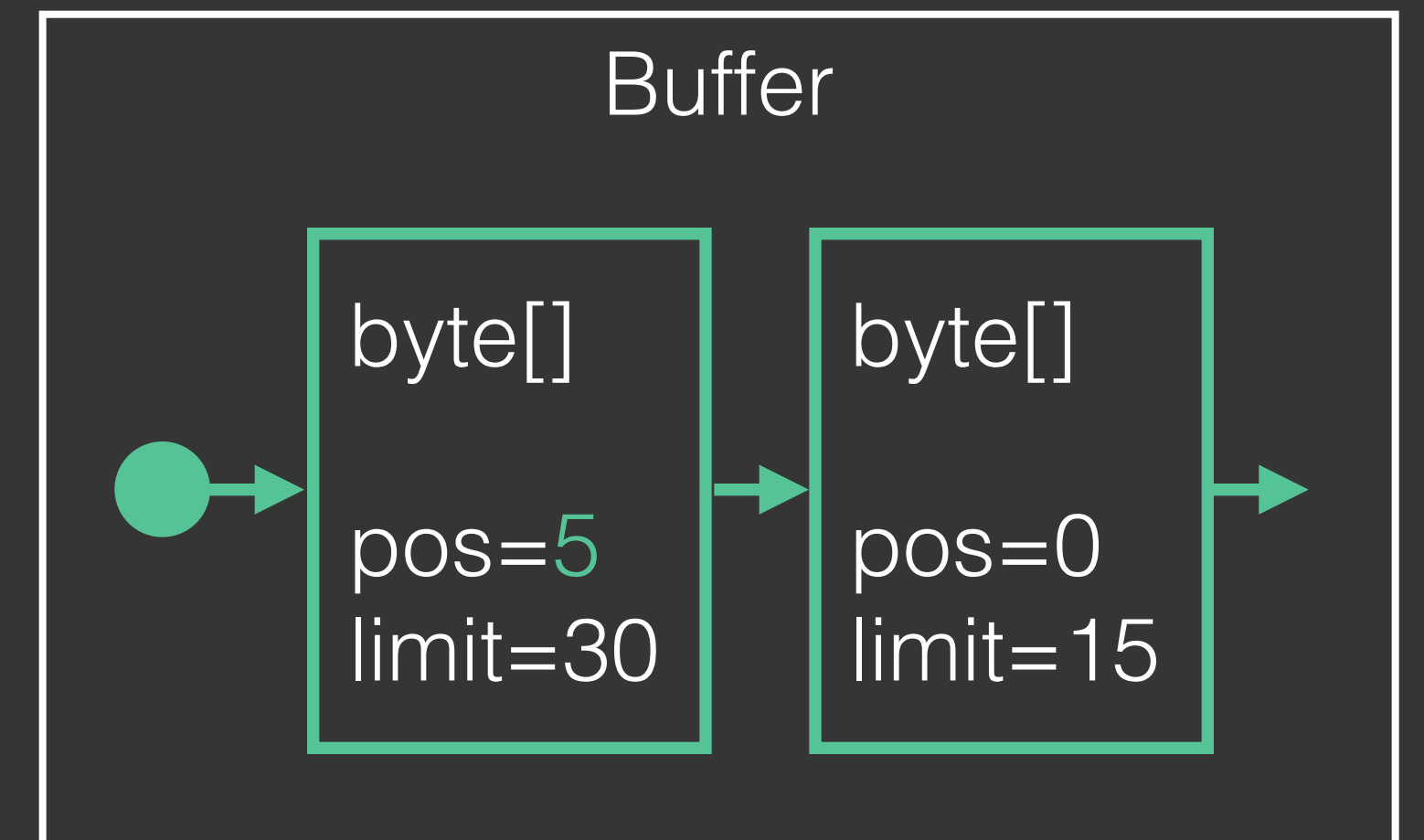
Okio

```
Buffer buffer = new Buffer();  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");  
  
buffer.readUtf8(5);
```



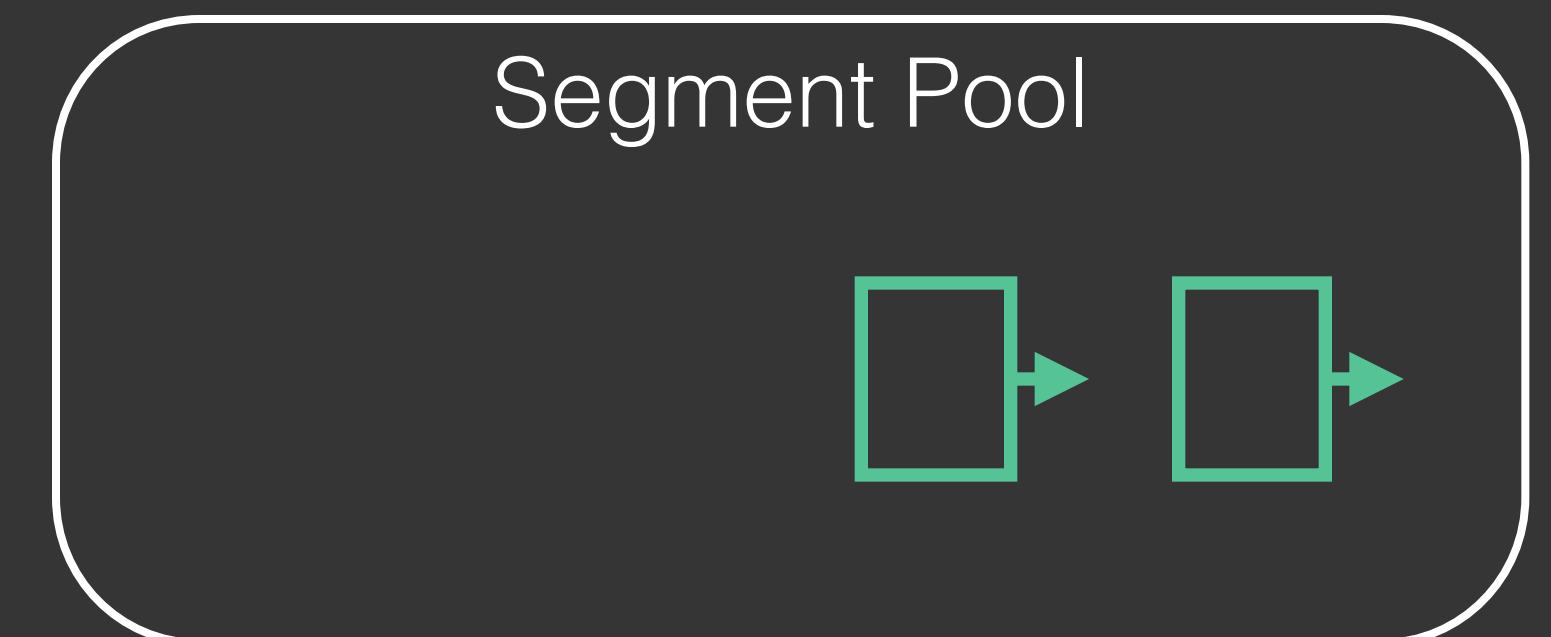
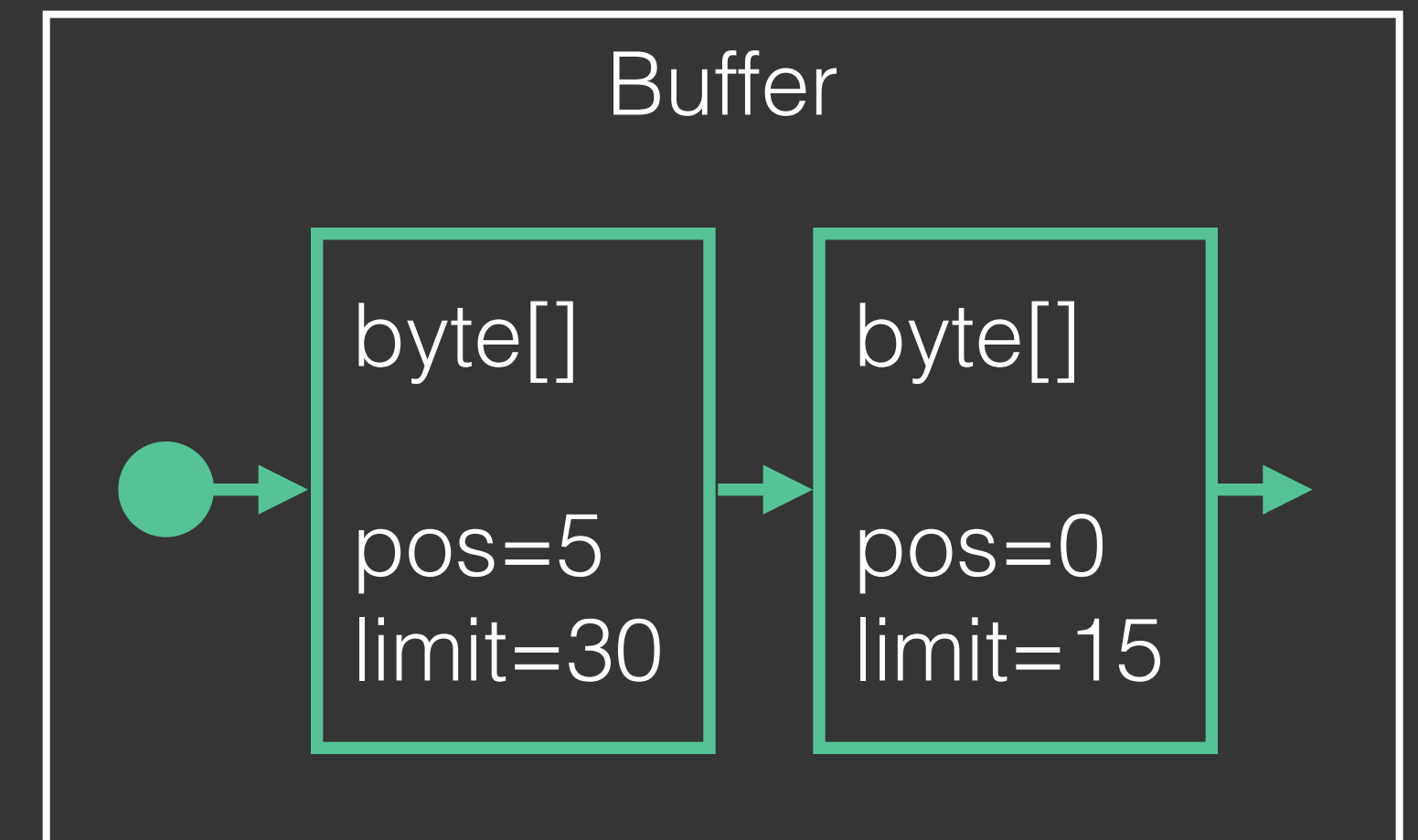
Okio

```
Buffer buffer = new Buffer();  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");  
  
buffer.readUtf8(5);  
// "Hello"
```



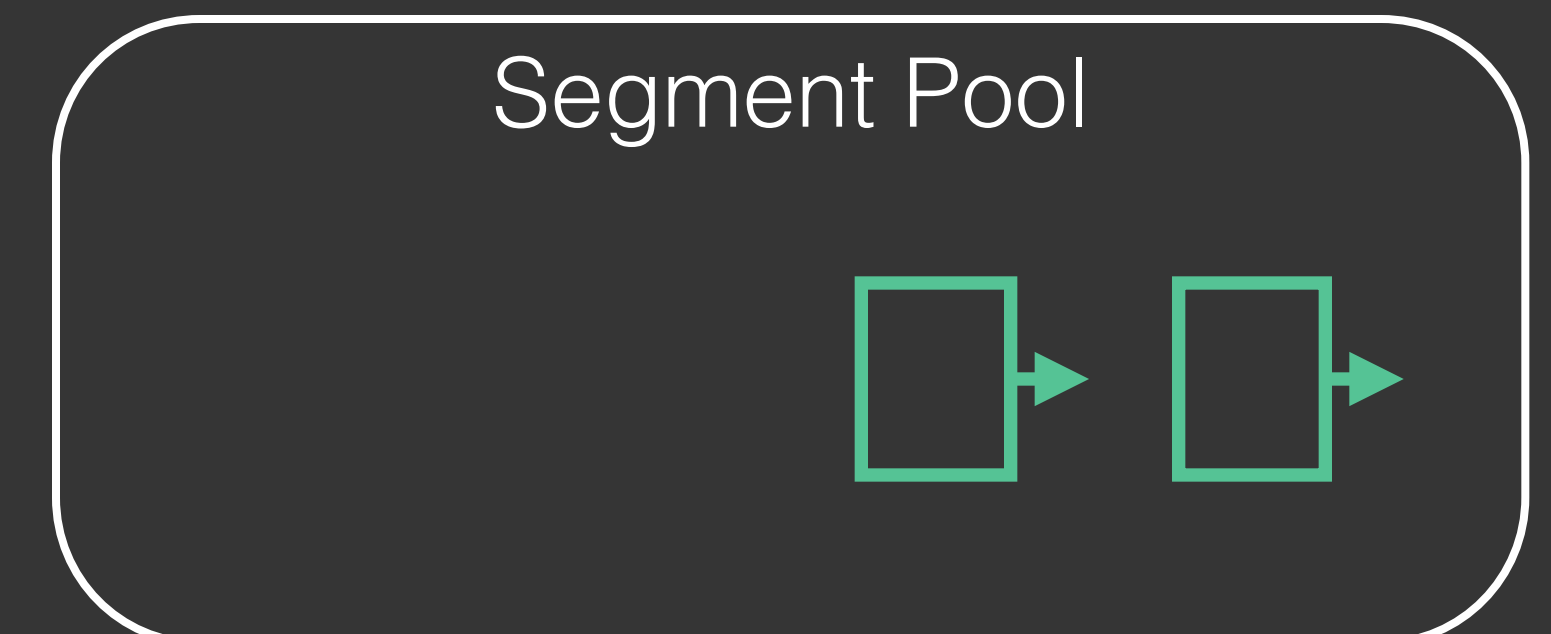
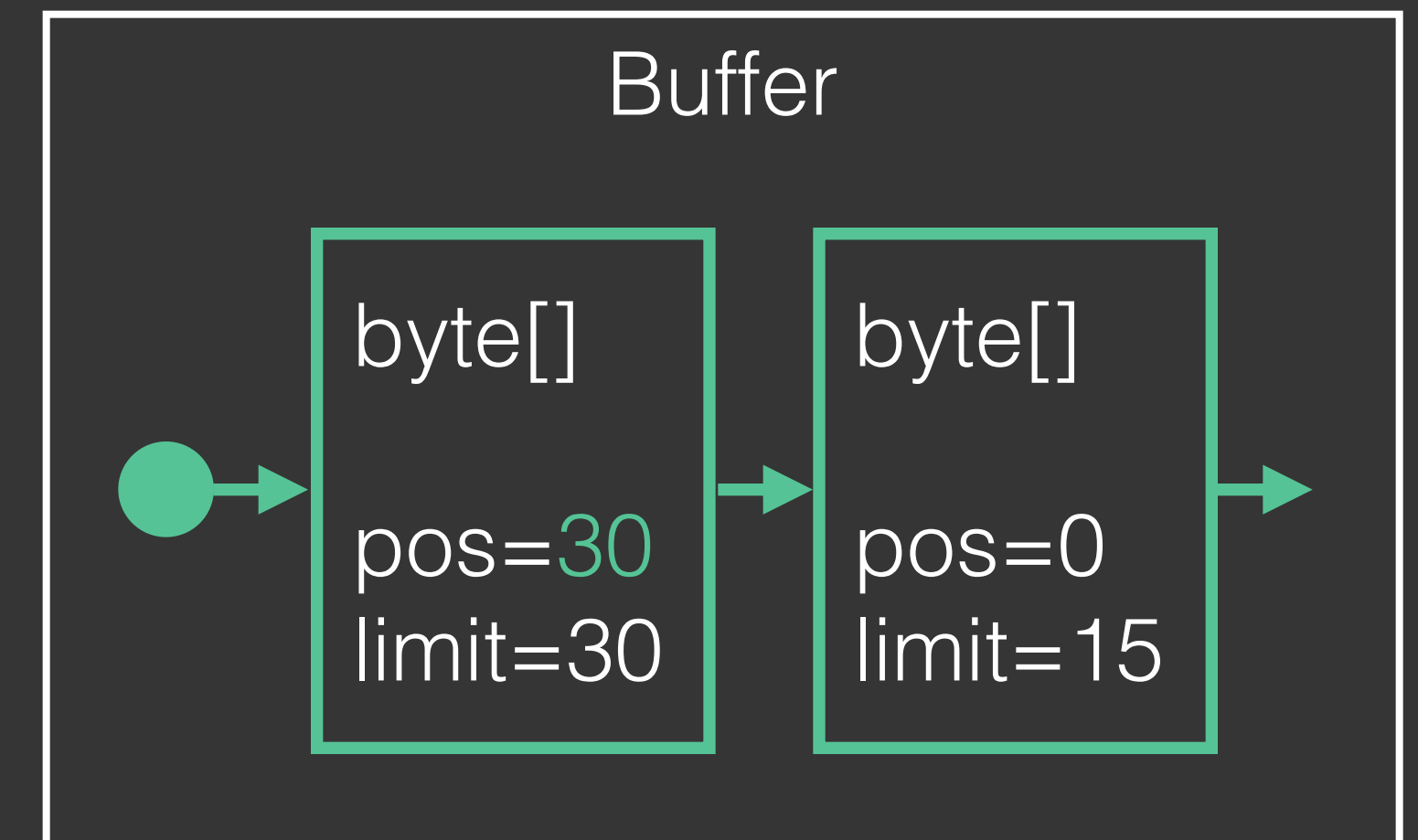
Okio

```
Buffer buffer = new Buffer();  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");  
  
buffer.readUtf8(5);  
// "Hello"  
buffer.readUtf8(25);
```



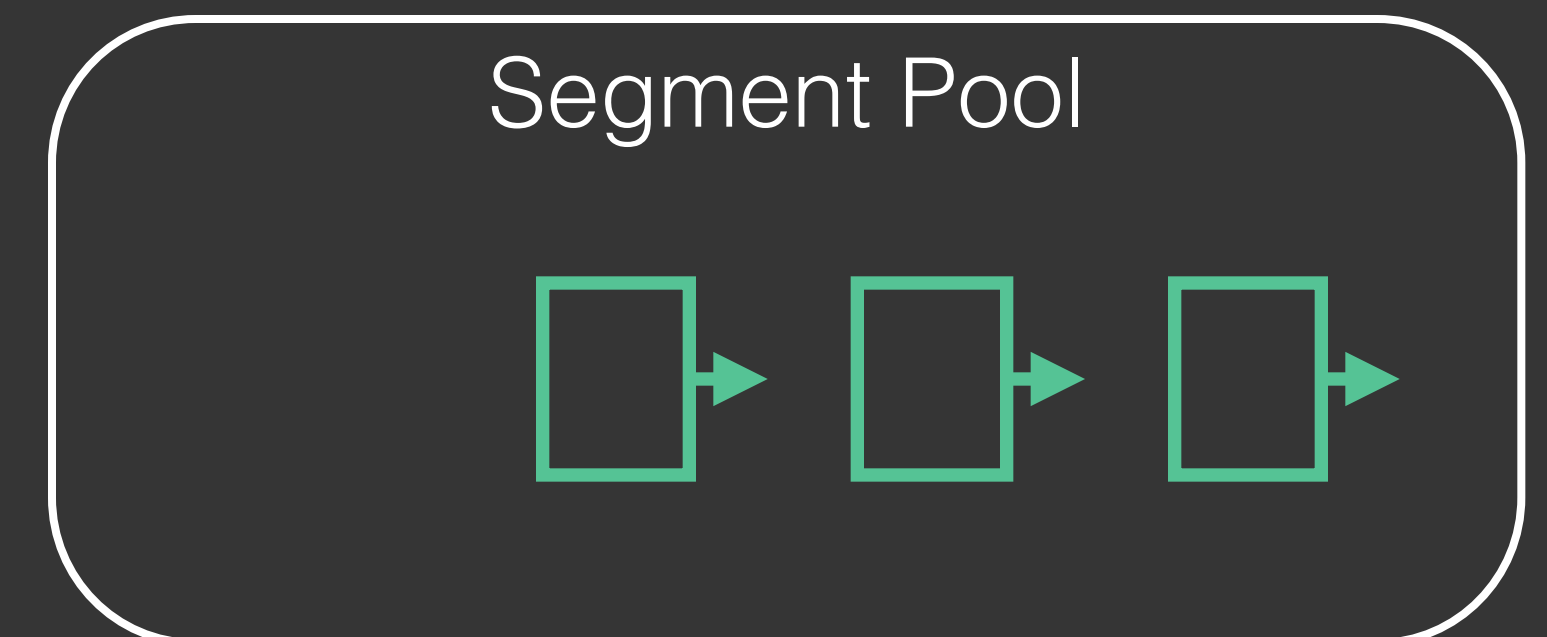
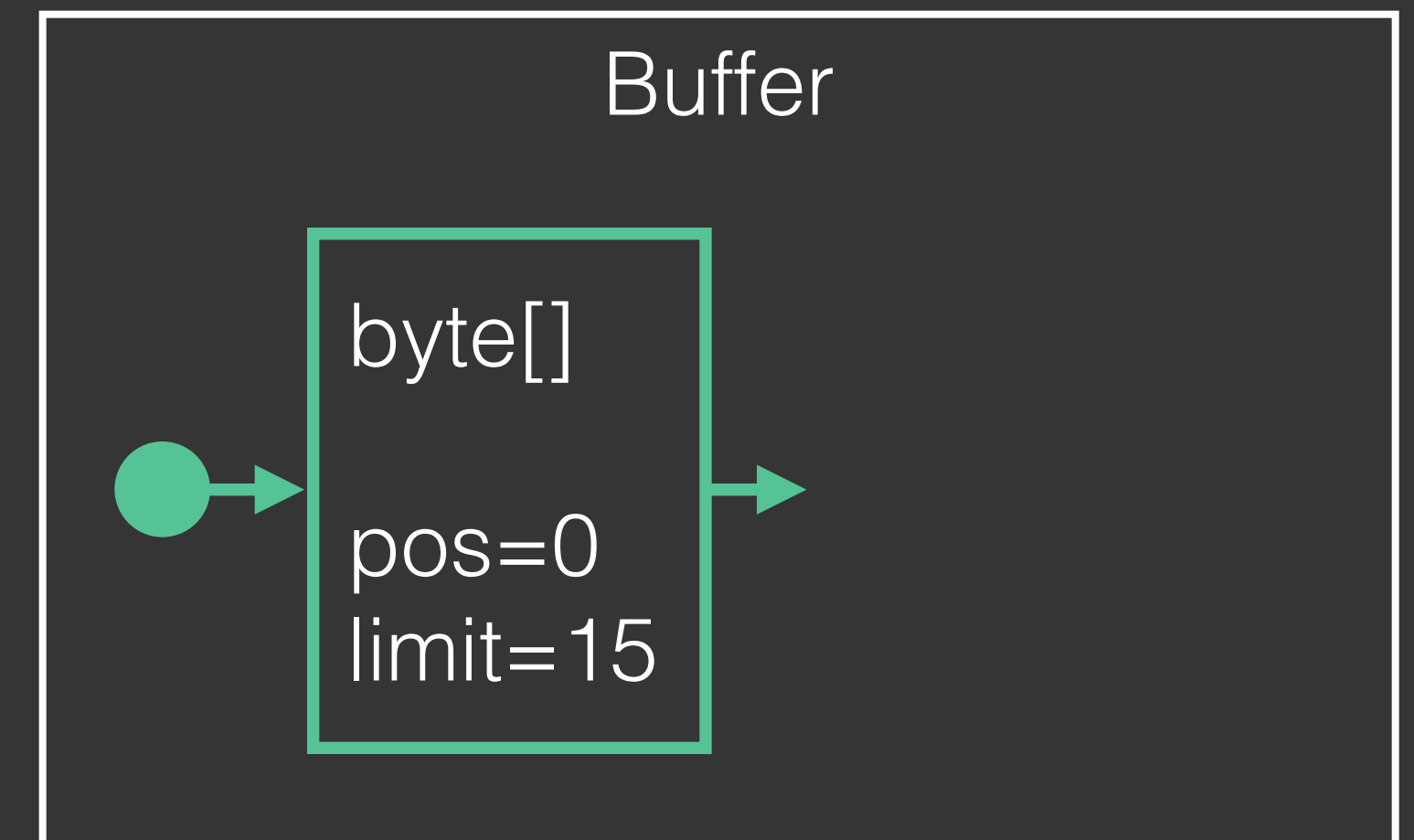
Okio

```
Buffer buffer = new Buffer();  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");  
  
buffer.readUtf8(5);  
// "Hello"  
buffer.readUtf8(25);  
// ", DroidconHello, Droidcon"
```



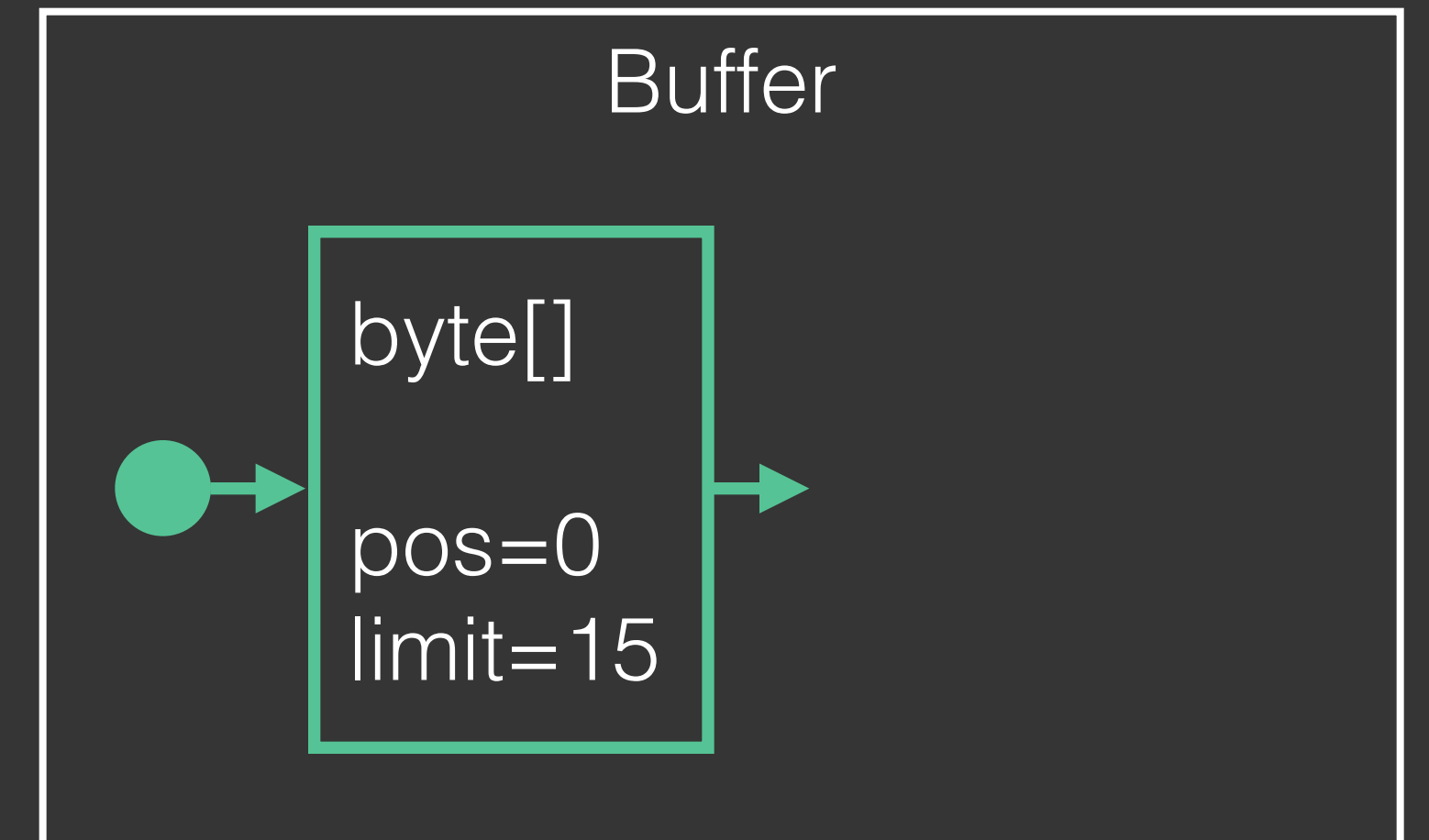
Okio

```
Buffer buffer = new Buffer();  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");  
  
buffer.readUtf8(5);  
// "Hello"  
buffer.readUtf8(25);  
// ", DroidconHello, Droidcon"
```



Okio

```
Buffer buffer = new Buffer();  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");  
  
buffer.readUtf8(5);  
// "Hello"  
buffer.readUtf8(25);  
// ", DroidconHello, Droidcon"  
  
Buffer otherBuffer = new Buffer();
```

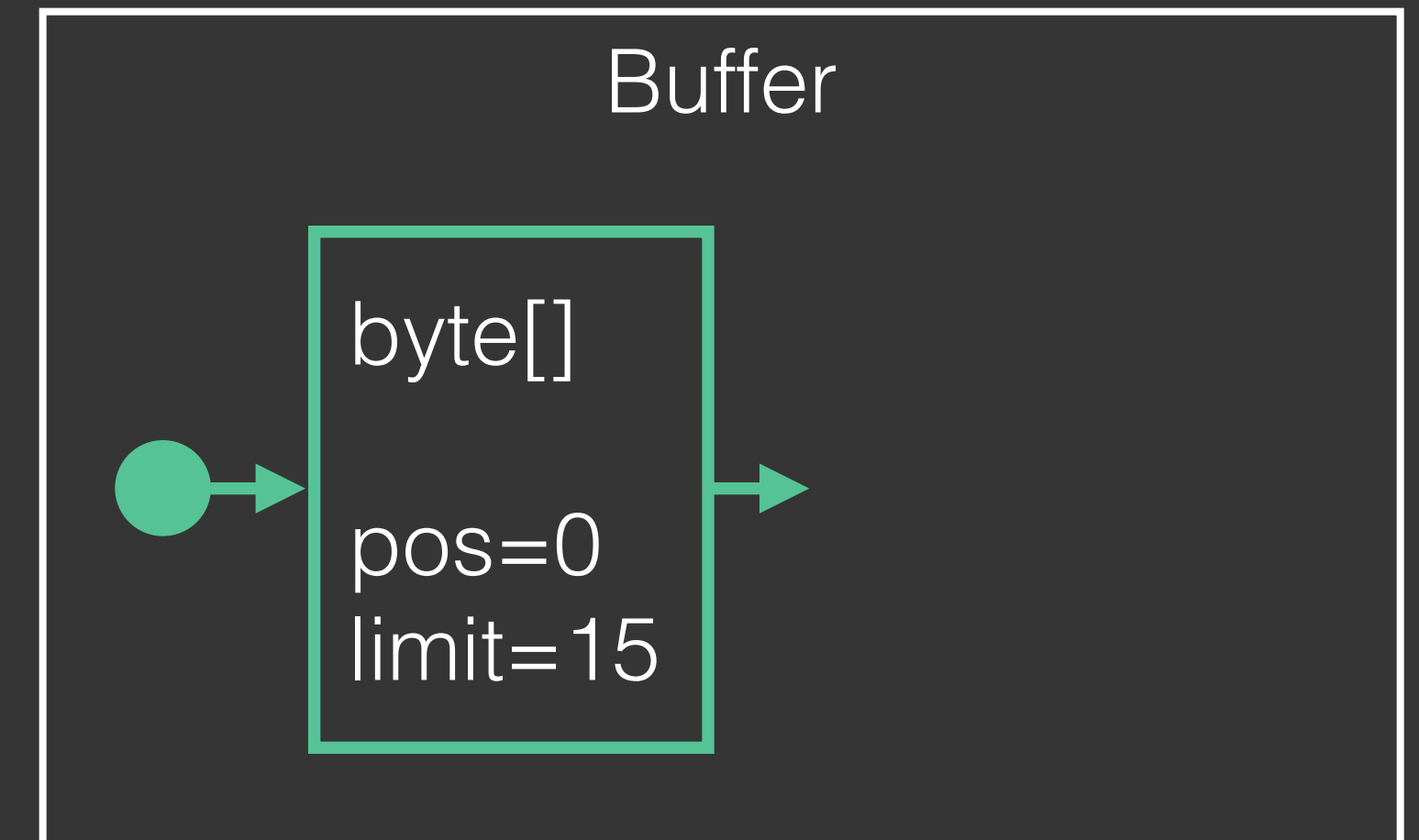


Okio

```
Buffer buffer = new Buffer();  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");
```

```
buffer.readUtf8(5);  
// "Hello"  
buffer.readUtf8(25);  
// ", DroidconHello, Droidcon"
```

```
Buffer otherBuffer = new Buffer();
```

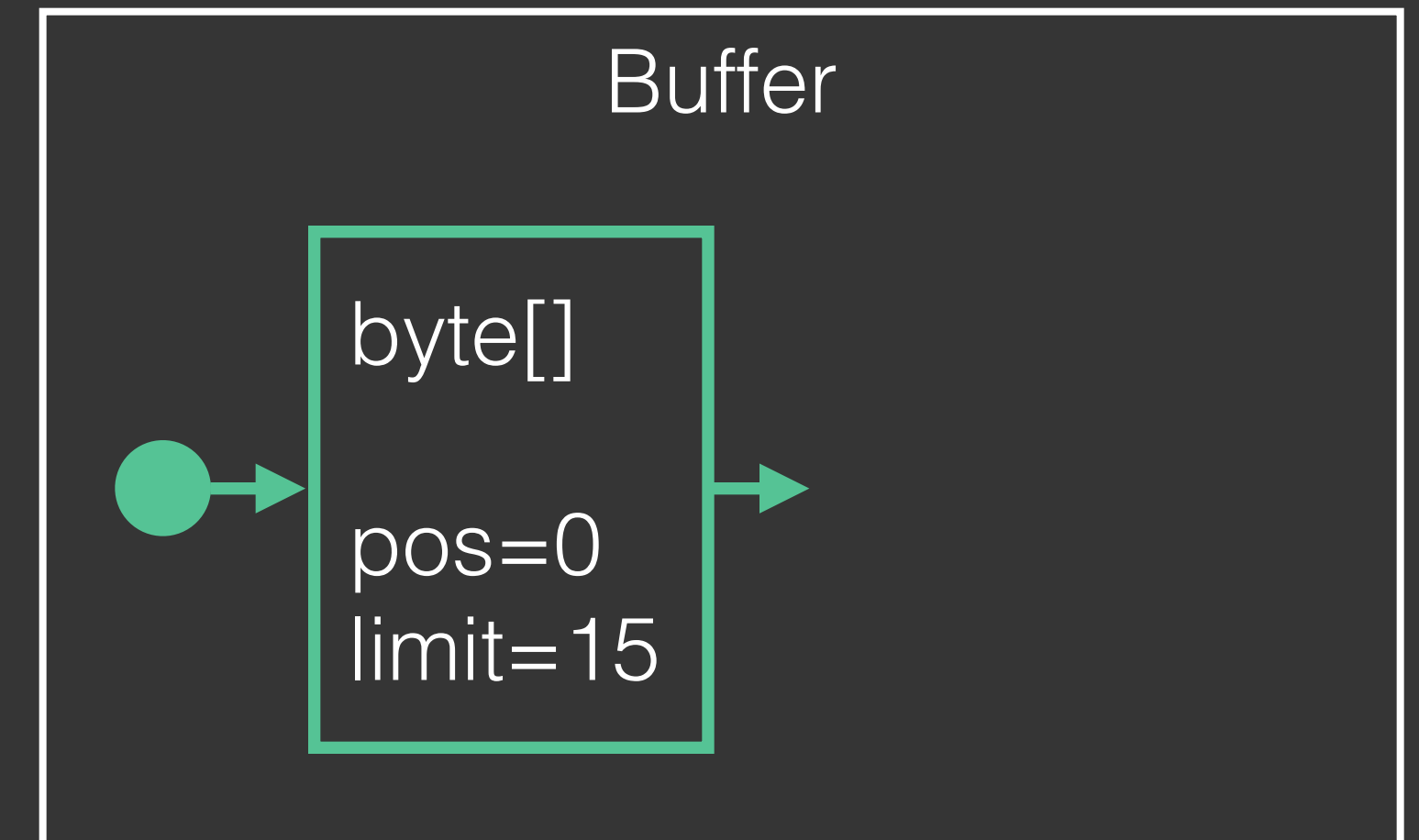


Okio

```
Buffer buffer = new Buffer();  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");
```

```
buffer.readUtf8(5);  
// "Hello"  
buffer.readUtf8(25);  
// ", DroidconHello, Droidcon"
```

```
Buffer otherBuffer = new Buffer();  
otherBuffer.writeAll(buffer);
```

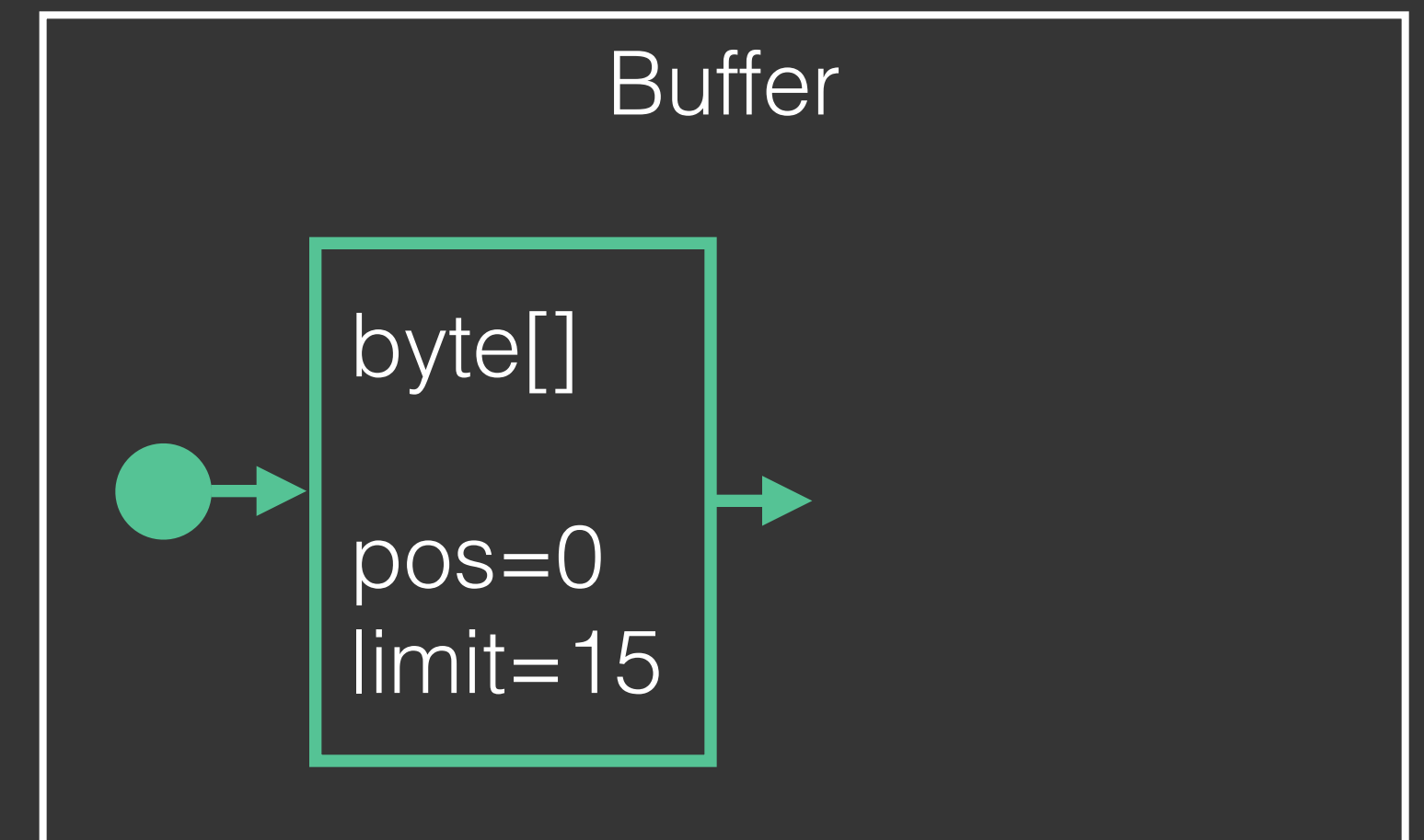


Okio

```
Buffer buffer = new Buffer();  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");  
buffer.writeUtf8("Hello, Droidcon");
```

```
buffer.readUtf8(5);  
// "Hello"  
buffer.readUtf8(25);  
// ", DroidconHello, Droidcon"
```

```
Buffer otherBuffer = new Buffer();  
otherBuffer.writeAll(buffer);
```



Okio

- Complements `java.io.*` and `java.nio.*`
- `Source` and `Sink` move data
- `Buffer` and `ByteString` hold data

Okio

- Complements `java.io.*` and `java.nio.*`
- `Source` and `Sink` move data
- `Buffer` and `ByteString` hold data
- `BufferedSource` and `BufferedSink` move data efficiently

Okio

```
interface BufferedSink extends Sink {  
}
```

Okio

```
interface BufferedSink extends Sink {  
}
```

```
interface BufferedSource extends Source {  
}
```

Okio

```
interface BufferedSink extends Sink {
    BufferedSink write(ByteString byteString);
    BufferedSink write(byte[] source);
    BufferedSink write(byte[] source, int offset, int byteCount);
    long writeAll(Source source);
    BufferedSink write(Source source, long byteCount);
    BufferedSink writeUtf8(String string);
    BufferedSink writeString(String string, Charset charset);
    BufferedSink writeByte(int b);
    BufferedSink writeShort(int s);
    BufferedSink writeShortLe(int s);
    BufferedSink writeInt(int i);
    BufferedSink writeIntLe(int i);
    BufferedSink writeLong(long v);
    BufferedSink writeLongLe(long v);
    BufferedSink writeDecimalLong(long v);
    BufferedSink writeHexadecimalUnsignedLong(long v);
}
```

```
interface BufferedSource extends Source {
}
```

Okio

```
interface BufferedSink extends Sink {
    BufferedSink write(ByteString byteString);
    BufferedSink write(byte[] source);
    BufferedSink write(byte[] source, int offset, int byteCount);
    long writeAll(Source source);
    BufferedSink write(Source source, long byteCount);
    BufferedSink writeUtf8(String string);
    BufferedSink writeString(String string, Charset charset);
    BufferedSink writeByte(int b);
    BufferedSink writeShort(int s);
    BufferedSink writeShortLe(int s);
    BufferedSink writeInt(int i);
    BufferedSink writeIntLe(int i);
    BufferedSink writeLong(long v);
    BufferedSink writeLongLe(long v);
    BufferedSink writeDecimalLong(long v);
    BufferedSink writeHexadecimalUnsignedLong(long v);
}
```

```
interface BufferedSource extends Source {
    byte readByte();
    short readShort();
    short readShortLe();
    int readInt();
    int readIntLe();
    long readLong();
    long readLongLe();
    long readDecimalLong();
    long readHexadecimalUnsignedLong();
    ByteString readByteString();
    ByteString readByteString(long byteCount);
    byte[] readByteArray();
    byte[] readByteArray(long byteCount);
    int read(byte[] sink);
    void readFully(byte[] sink);
    int read(byte[] sink, int offset, int byteCount);
    void readFully(Buffer sink, long byteCount);
    long readAll(Sink sink);
    String readUtf8();
    String readUtf8(long byteCount);
    String readUtf8Line();
    String readUtf8LineStrict();
    String readString(Charset charset);
    String readString(long byteCount, Charset charset);

    long indexOf(byte b);
    long indexOf(byte b, long fromIndex);
    long indexOfElement(ByteString targetBytes);
    long indexOfElement(ByteString targetBytes, long fromIndex);
}
```

Okio

```
interface BufferedSink extends Sink {
    BufferedSink write(ByteString byteString);
    BufferedSink write(byte[] source);
    BufferedSink write(byte[] source, int offset, int byteCount);
    long writeAll(Source source);
    BufferedSink write(Source source, long byteCount);
    BufferedSink writeUtf8(String string);
    BufferedSink writeString(String string, Charset charset);
    BufferedSink writeByte(int b);
    BufferedSink writeShort(int s);
    BufferedSink writeShortLe(int s);
    BufferedSink writeInt(int i);
    BufferedSink writeIntLe(int i);
    BufferedSink writeLong(long v);
    BufferedSink writeLongLe(long v);
    BufferedSink writeDecimalLong(long v);
    BufferedSink writeHexadecimalUnsignedLong(long v);

    BufferedSink emitCompleteSegments();
    BufferedSink emit();

    Buffer buffer();
    OutputStream outputStream();
}
```

```
interface BufferedSource extends Source {
    byte readByte();
    short readShort();
    short readShortLe();
    int readInt();
    int readIntLe();
    long readLong();
    long readLongLe();
    long readDecimalLong();
    long readHexadecimalUnsignedLong();
    ByteString readByteString();
    ByteString readByteString(long byteCount);
    byte[] readByteArray();
    byte[] readByteArray(long byteCount);
    int read(byte[] sink);
    void readFully(byte[] sink);
    int read(byte[] sink, int offset, int byteCount);
    void readFully(Buffer sink, long byteCount);
    long readAll(Sink sink);
    String readUtf8();
    String readUtf8(long byteCount);
    String readUtf8Line();
    String readUtf8LineStrict();
    String readString(Charset charset);
    String readString(long byteCount, Charset charset);

    long indexOf(byte b);
    long indexOf(byte b, long fromIndex);
    long indexOfElement(ByteString targetBytes);
    long indexOfElement(ByteString targetBytes, long fromIndex);

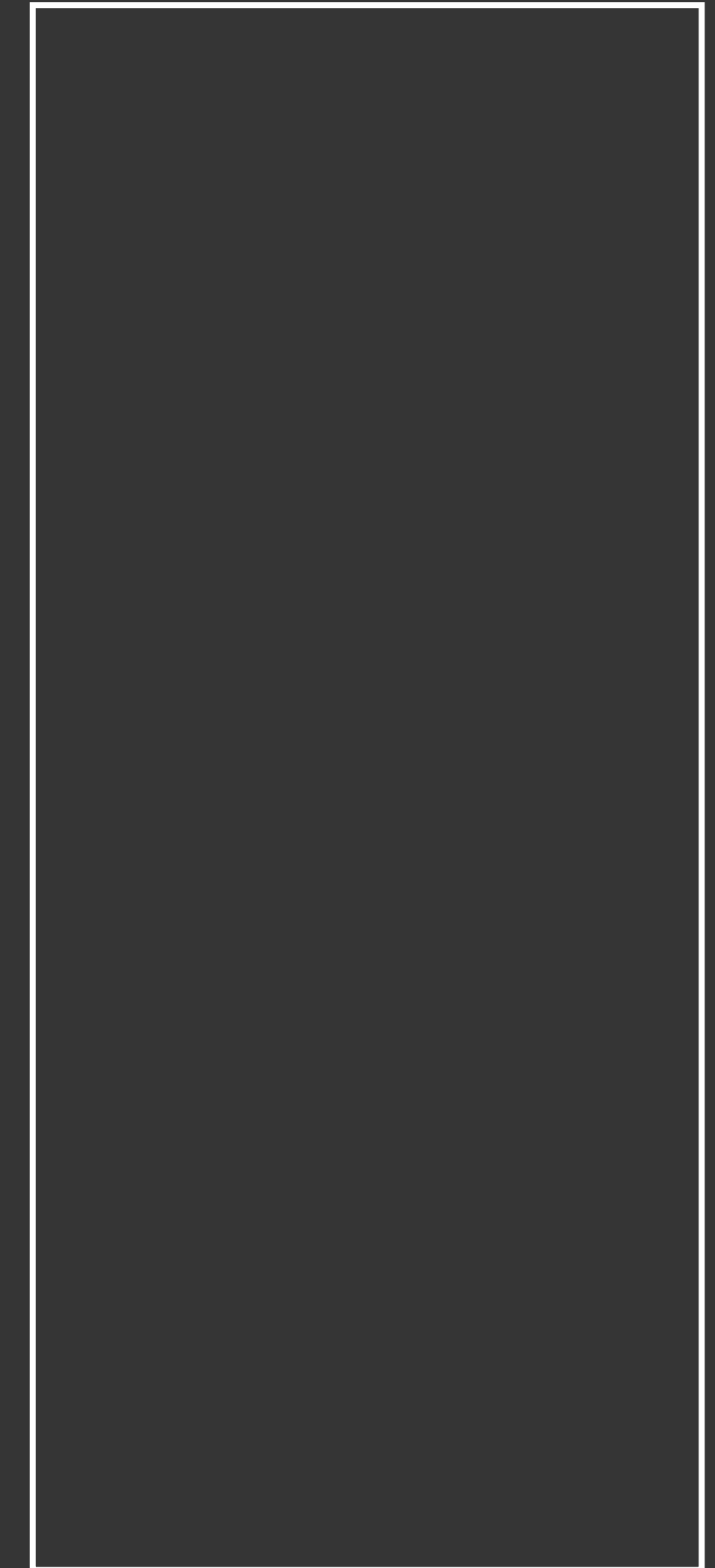
    boolean exhausted();
    void require(long byteCount);
    boolean request(long byteCount);
    void skip(long byteCount);

    Buffer buffer();
    InputStream inputStream();
}
```


Okio

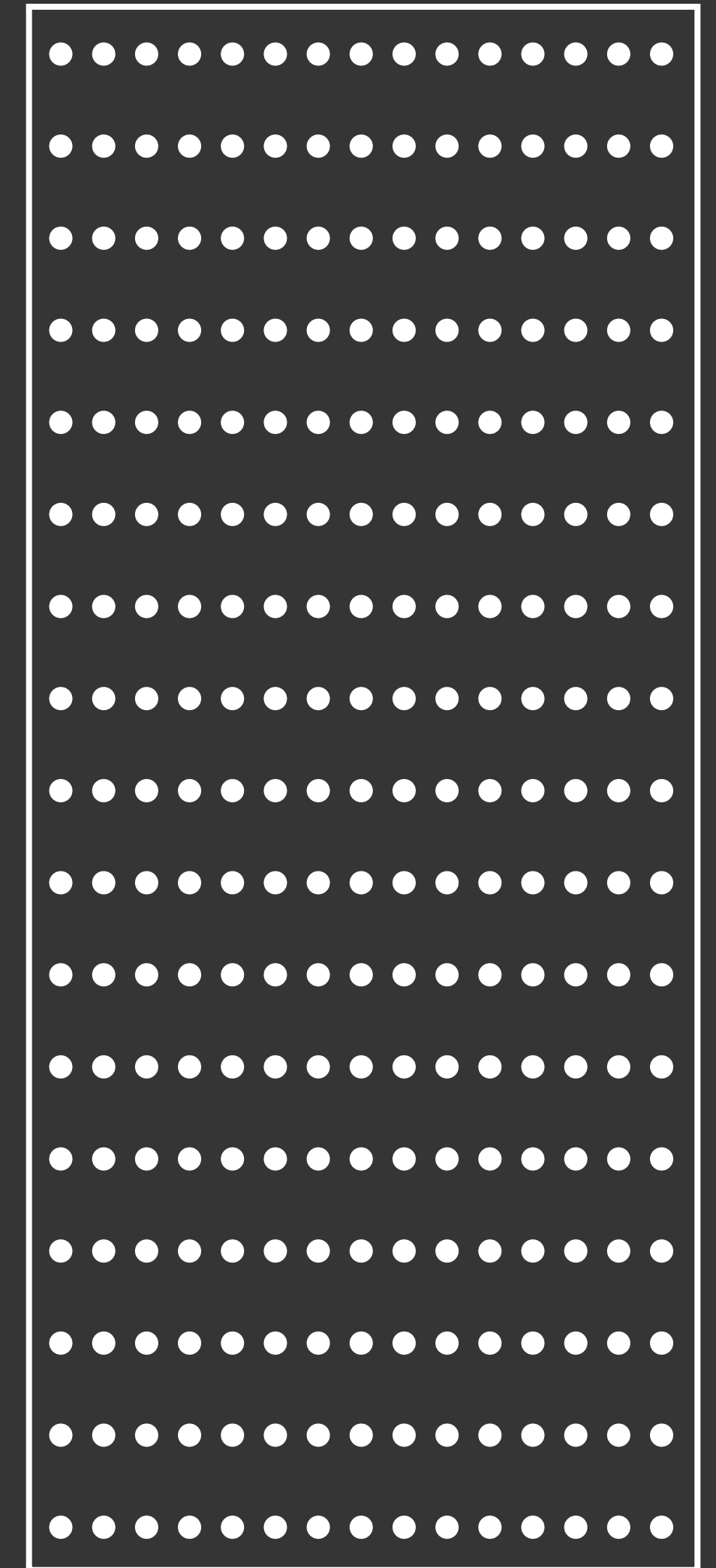
```
class Buffer implements BufferedSink, BufferedSource {  
}
```

Okio



Socket

Okio

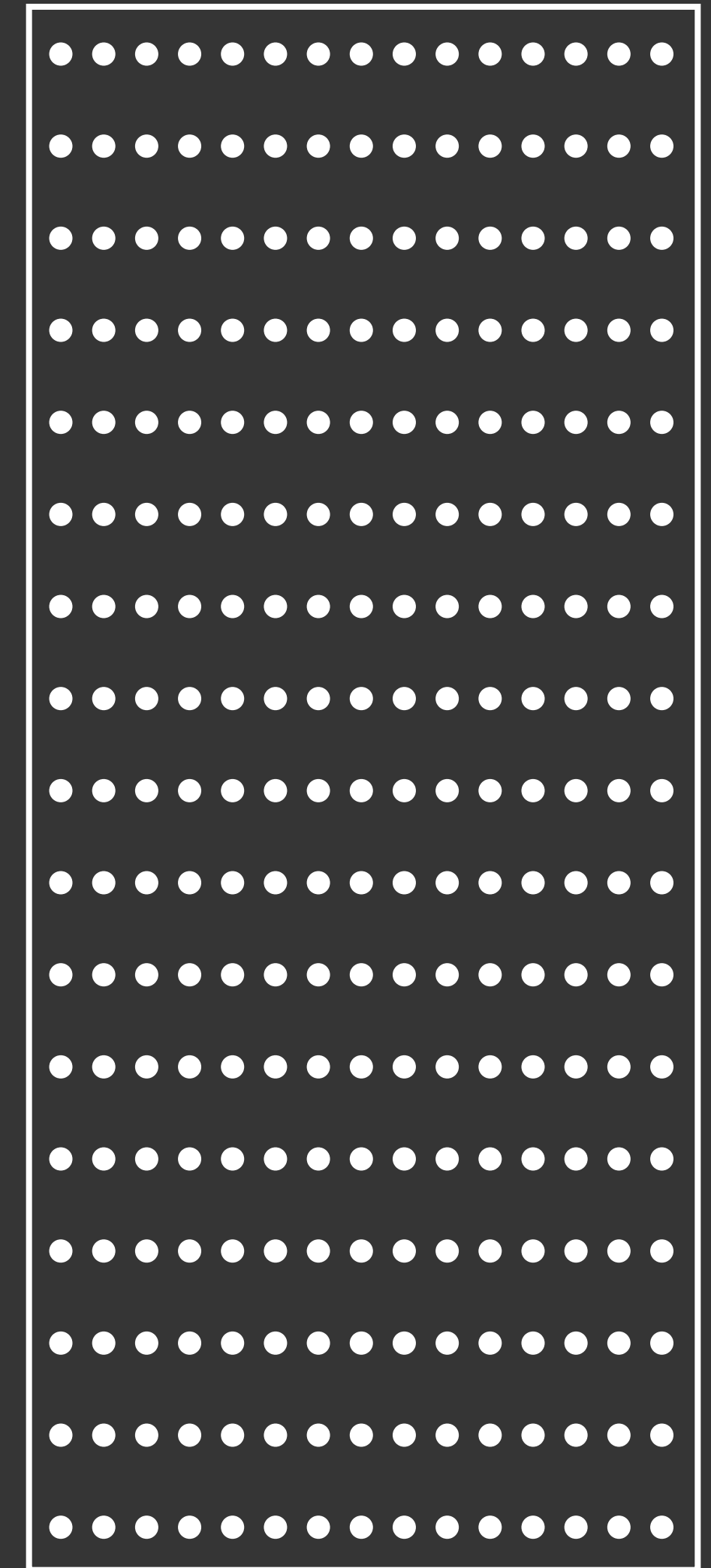


Socket

Okio



BufferedSource



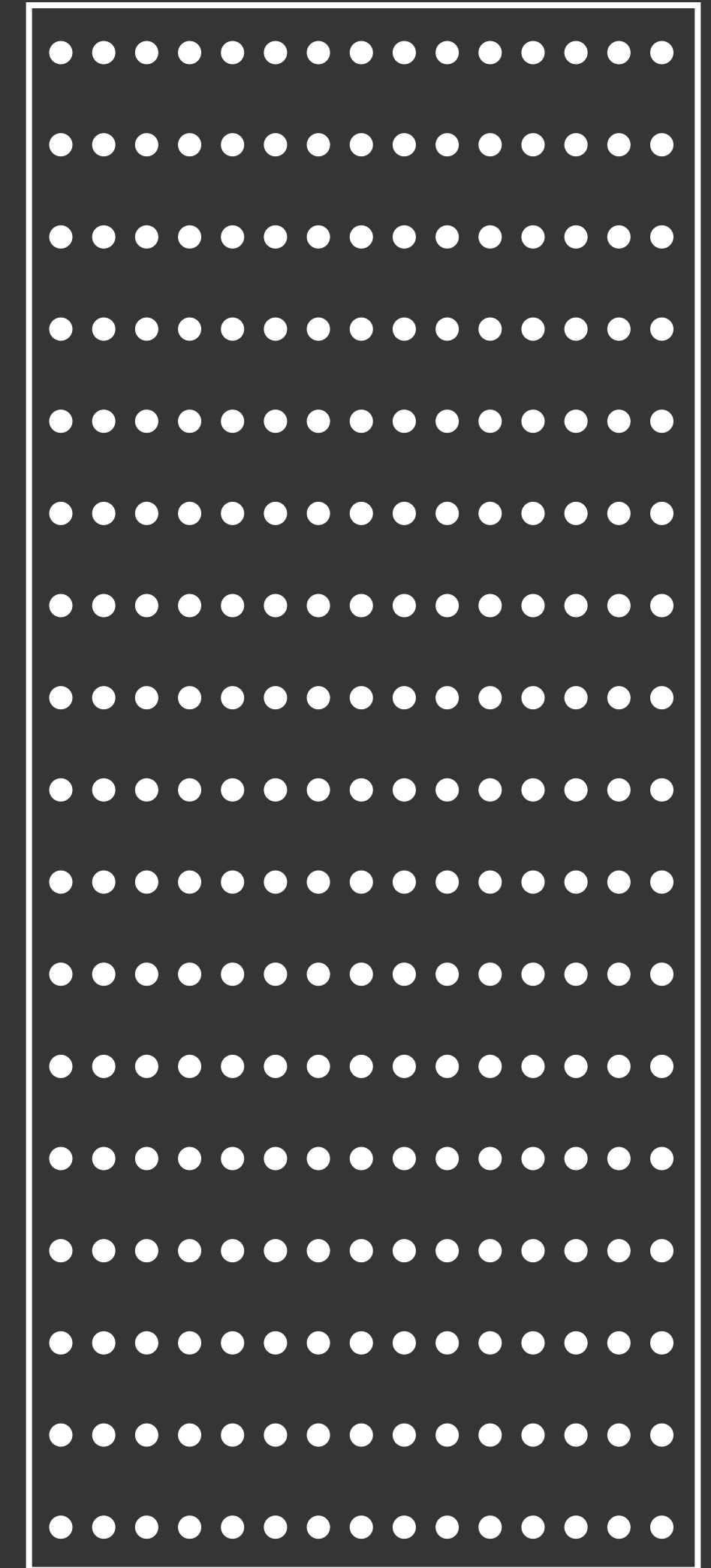
Socket

Okio

```
source.readInt();
```



BufferedSource



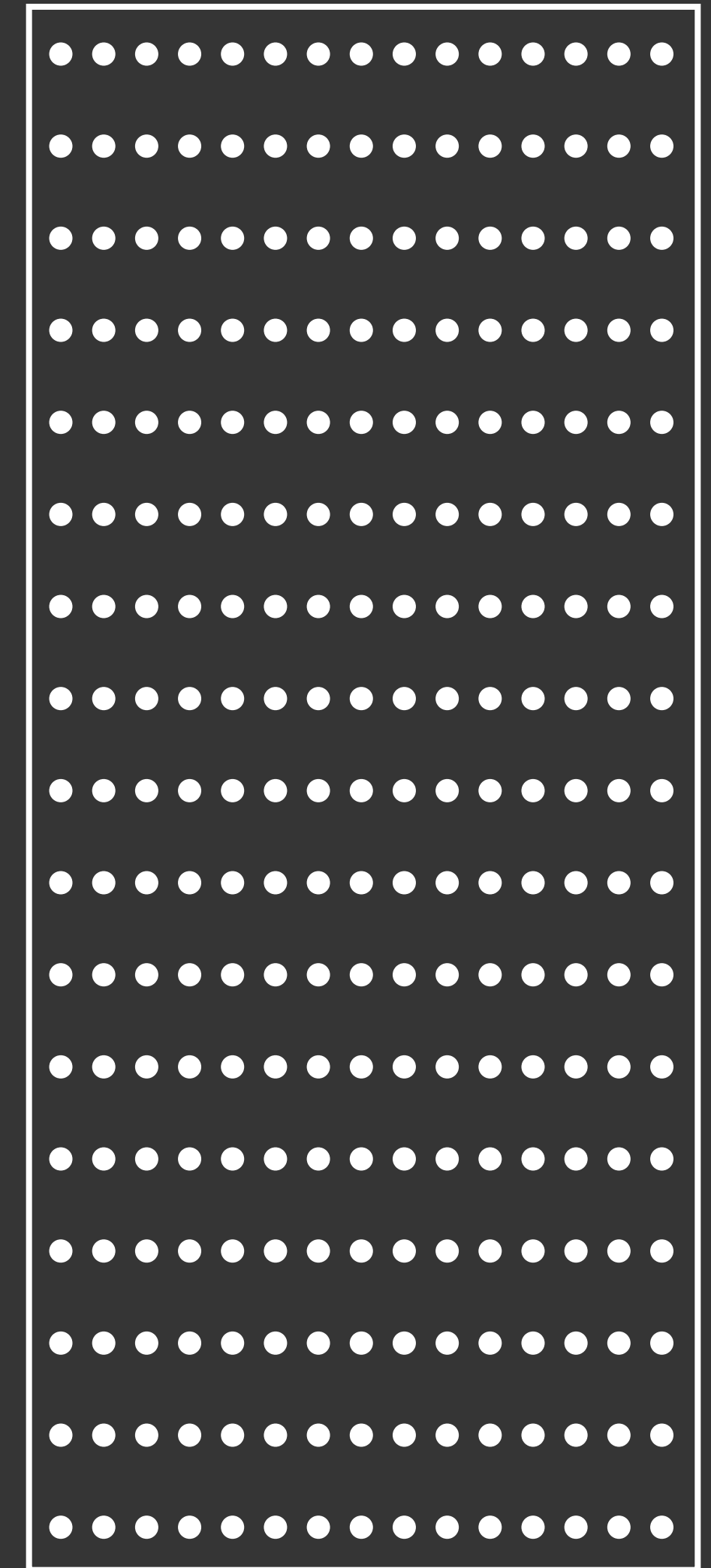
Socket

Okio

```
source.readInt();
```



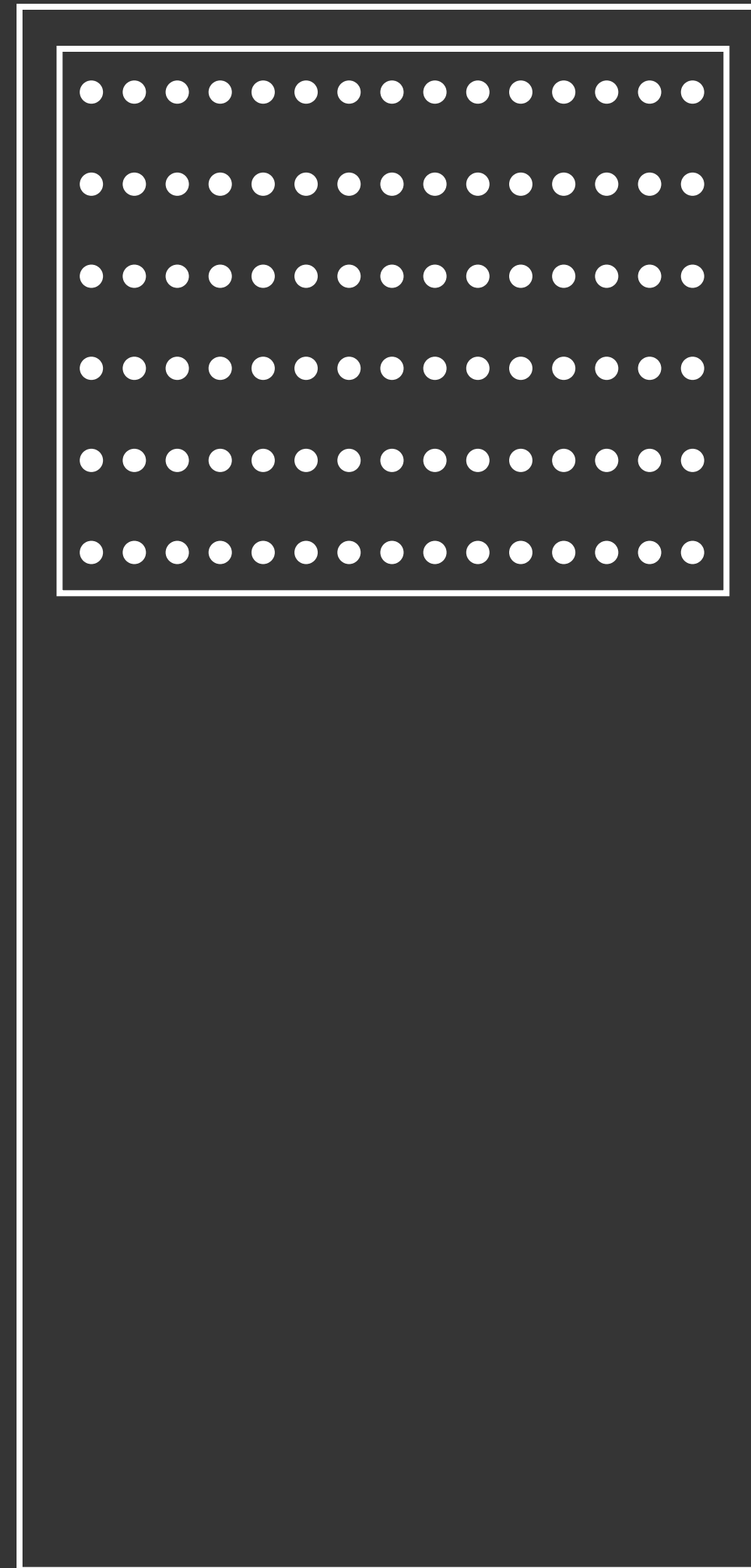
BufferedSource



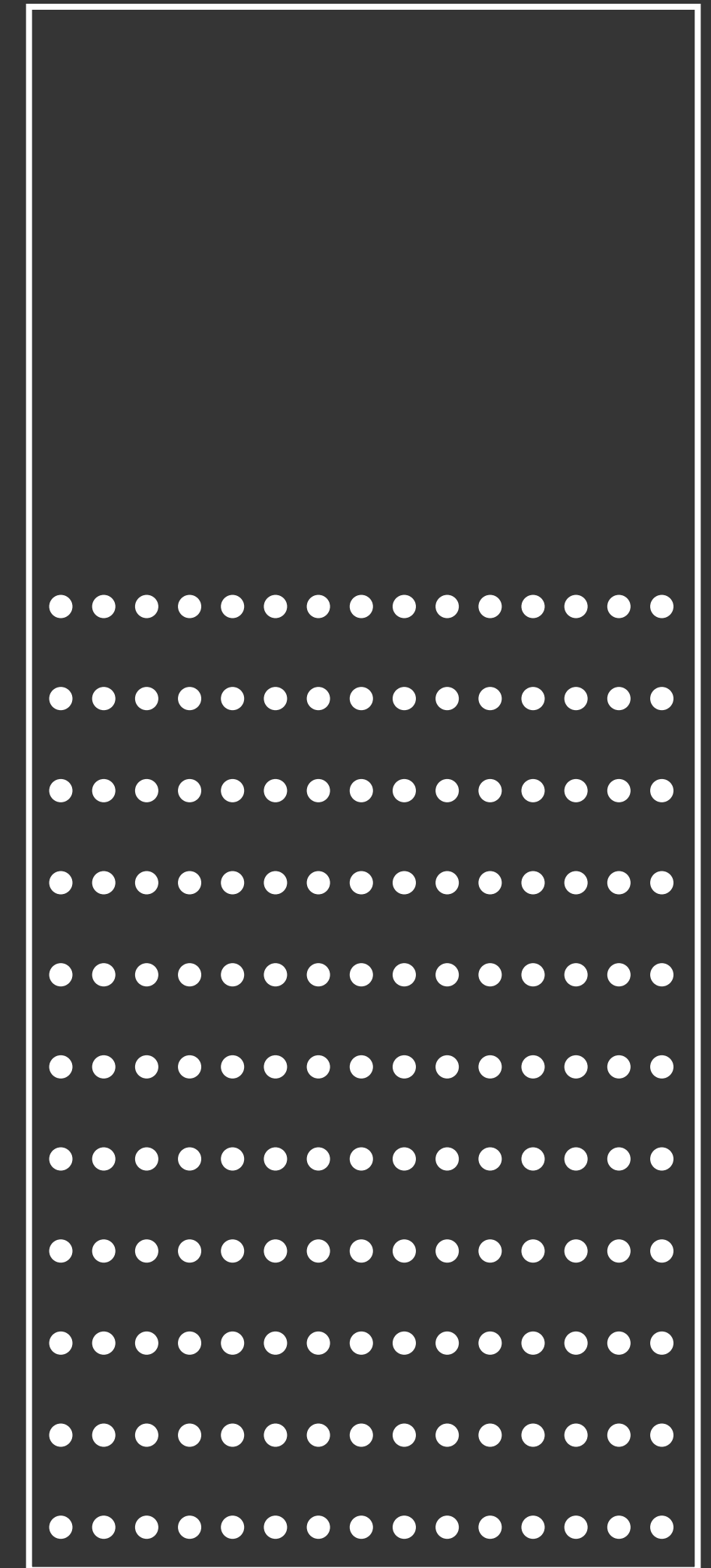
Socket

Okio

```
source.readInt();
```



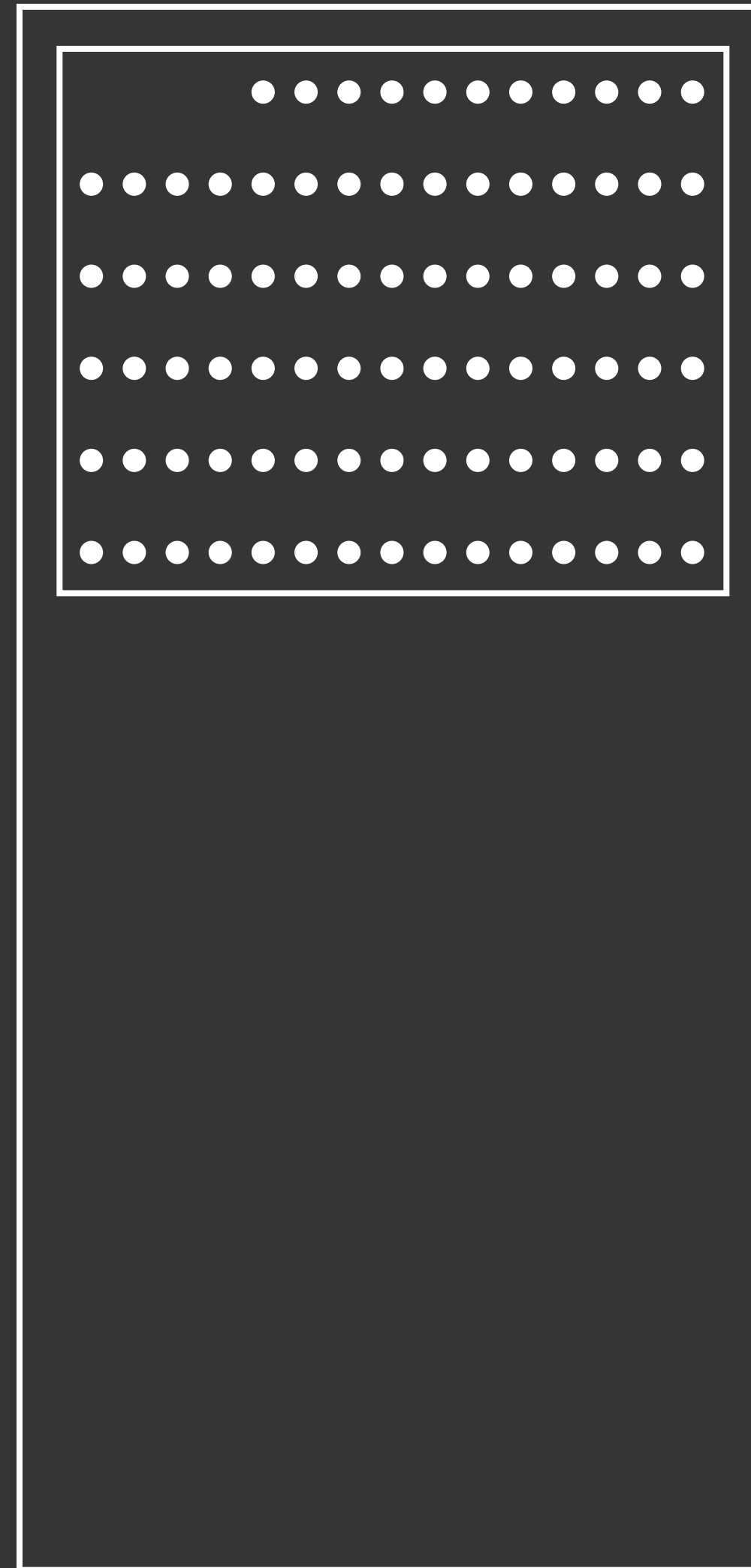
BufferedSource



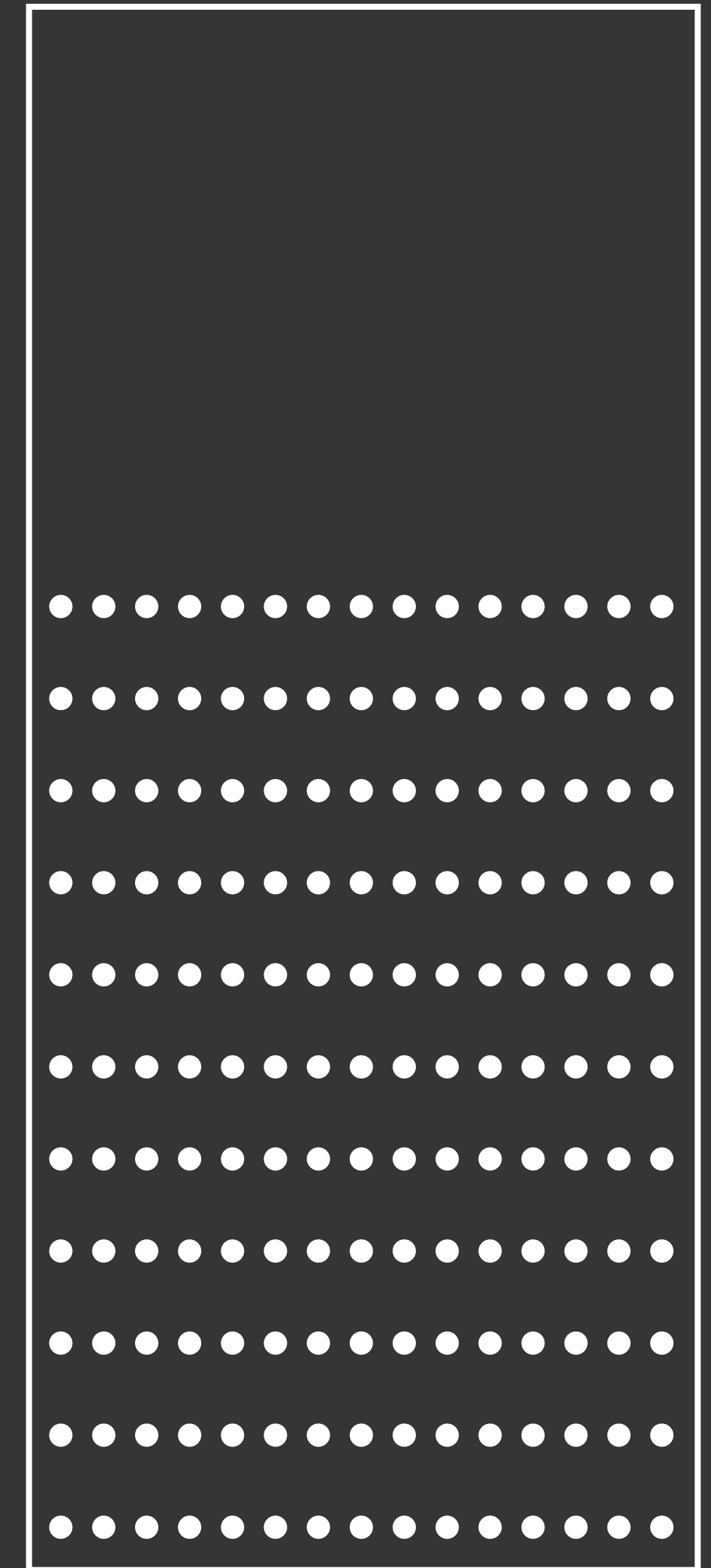
Socket

Okio

```
source.readInt(); ....
```



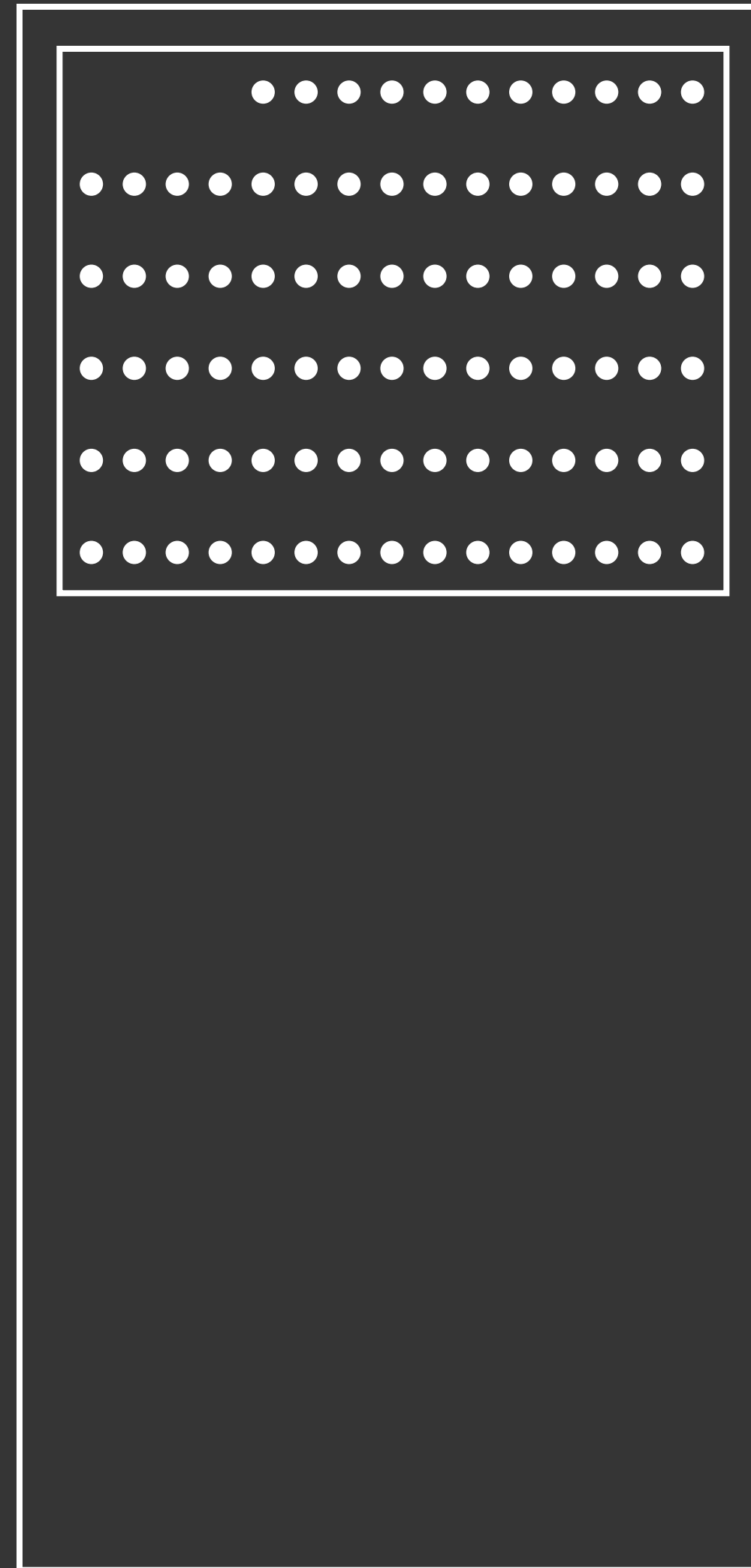
BufferedSource



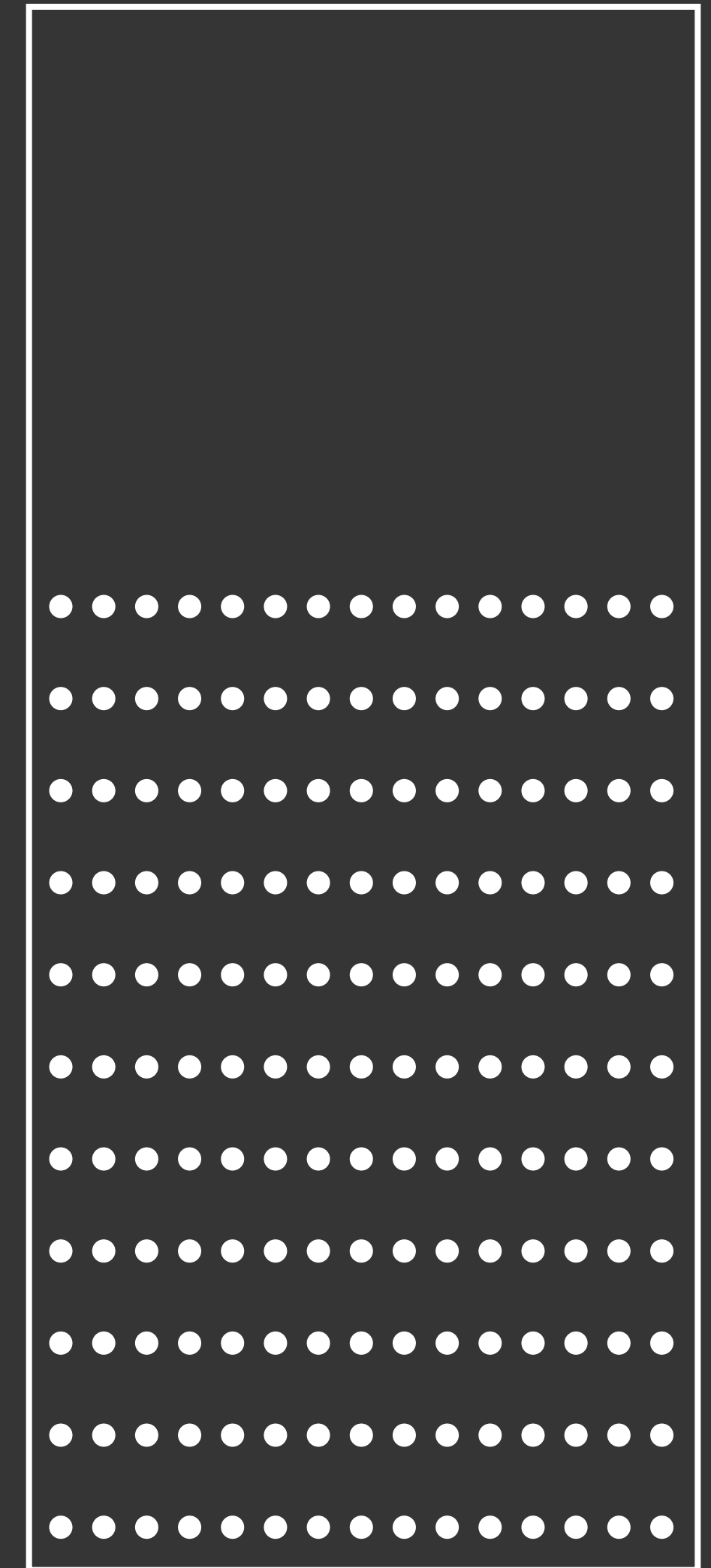
Socket

Okio

```
source.readInt(); ....  
source.readInt();
```



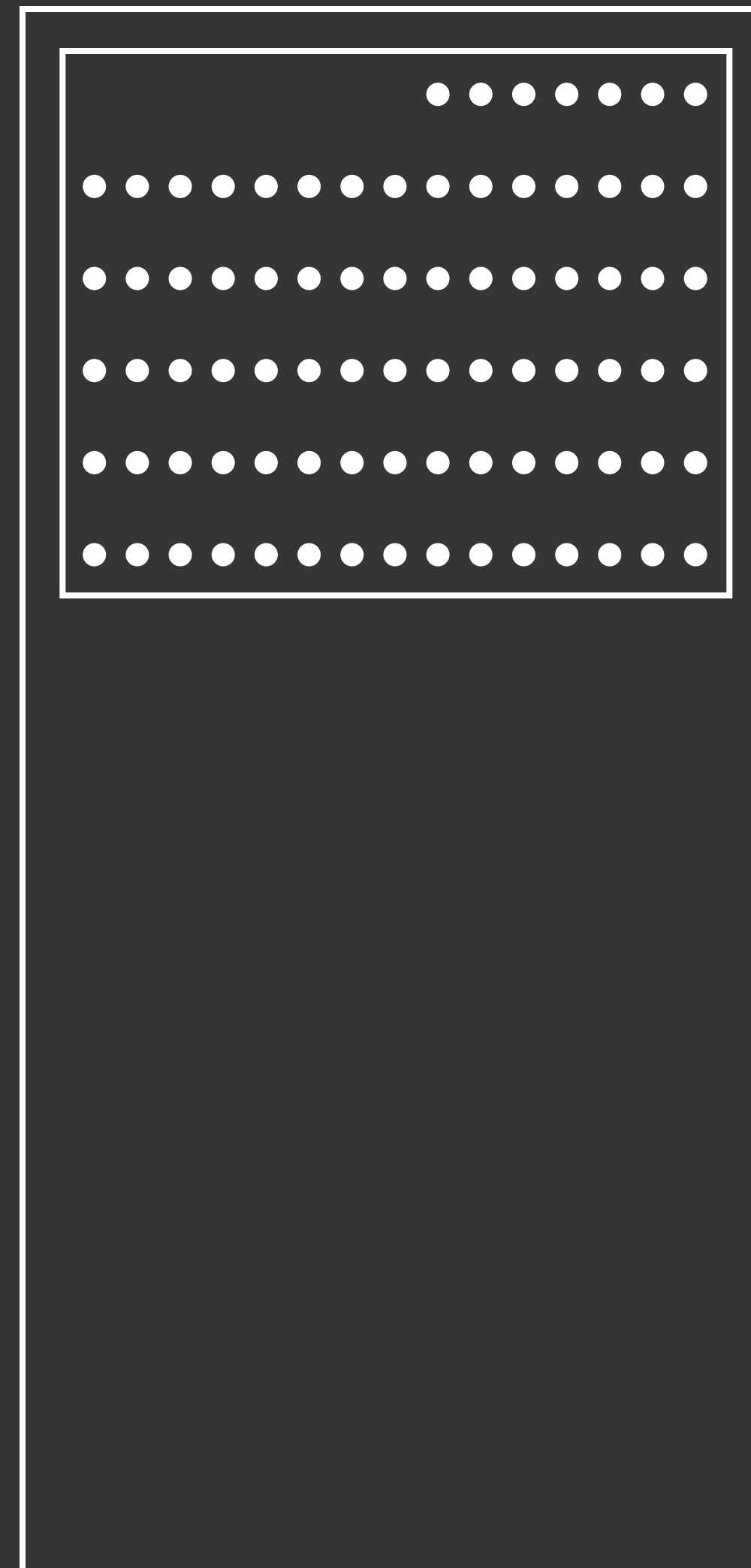
BufferedSource



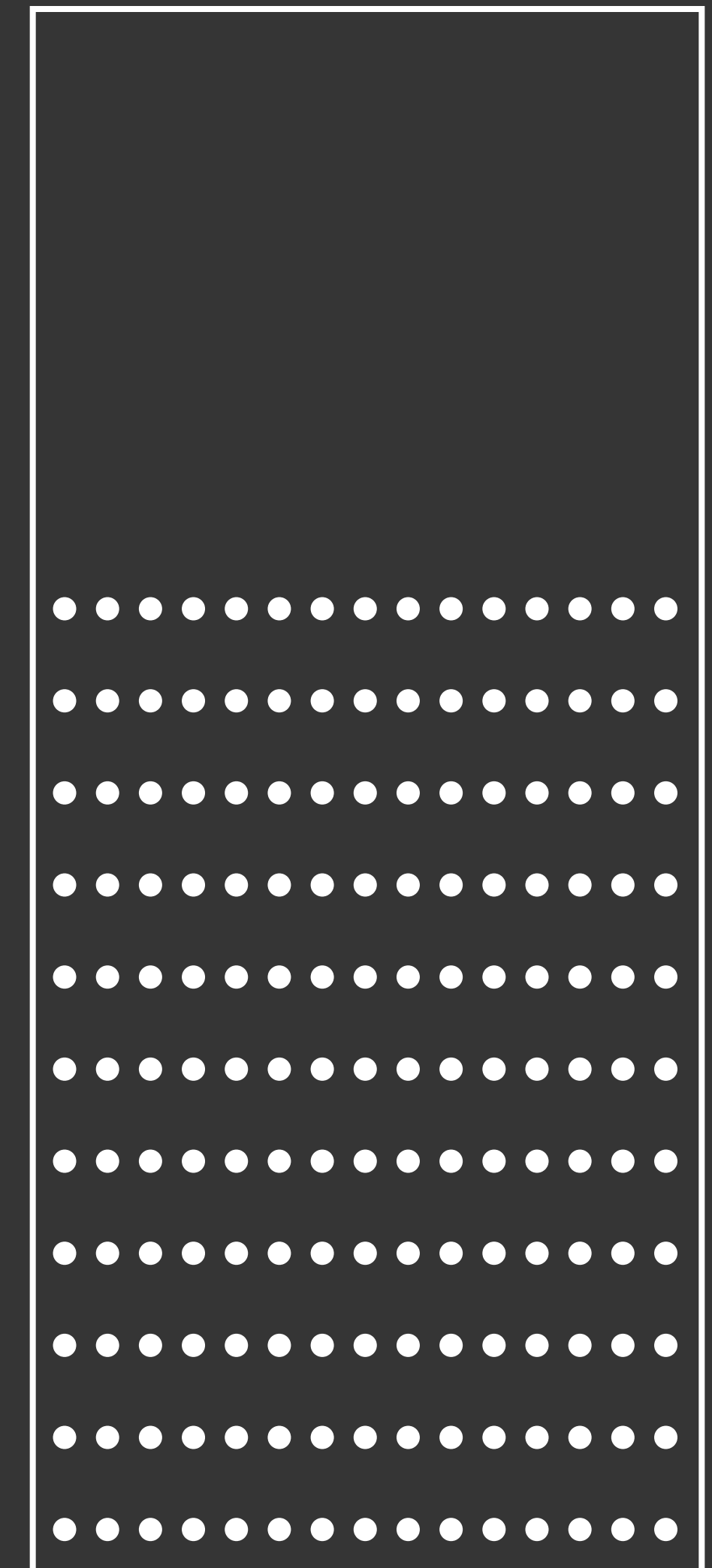
Socket

Okio

```
source.readInt(); ••••
source.readInt(); ••••
```



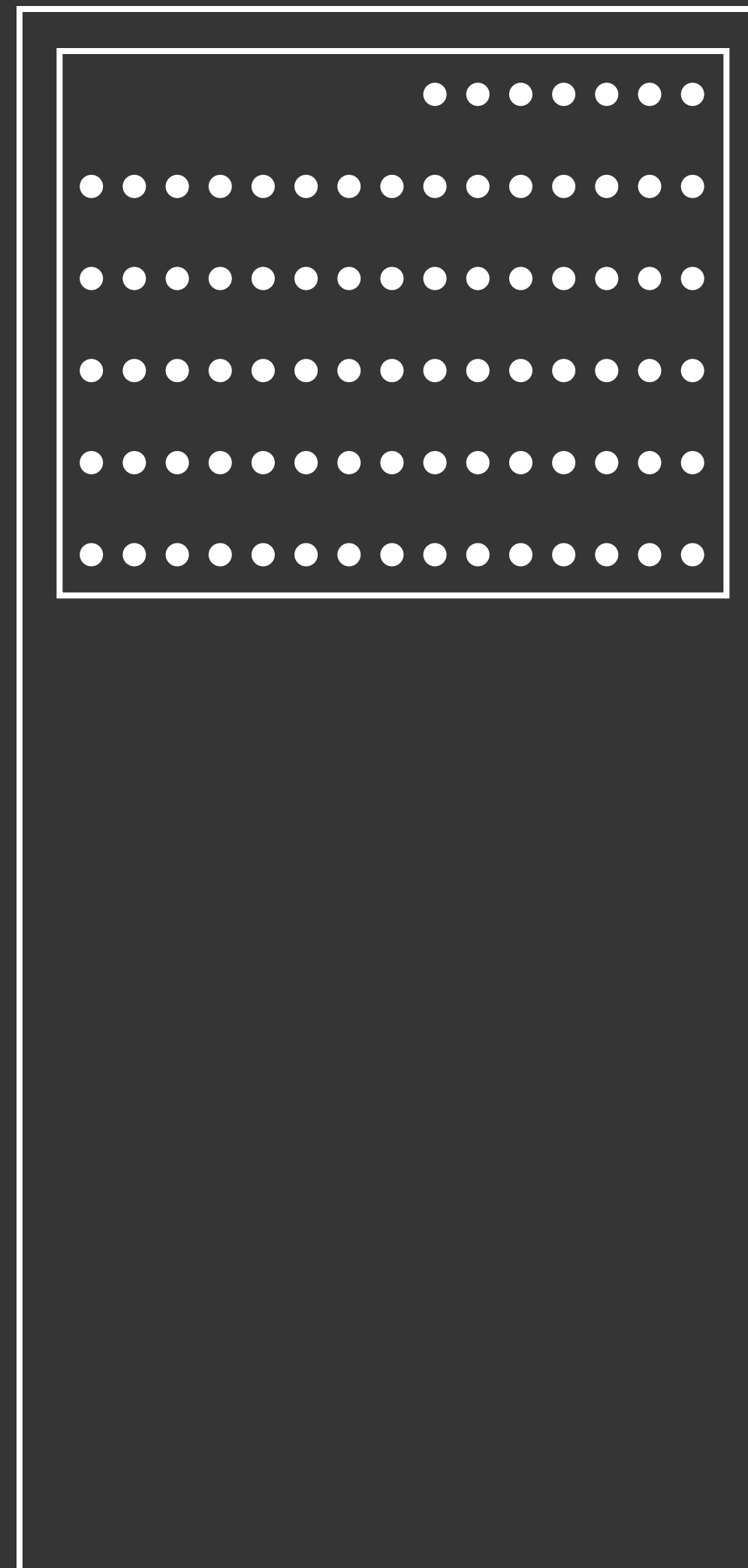
BufferedSource



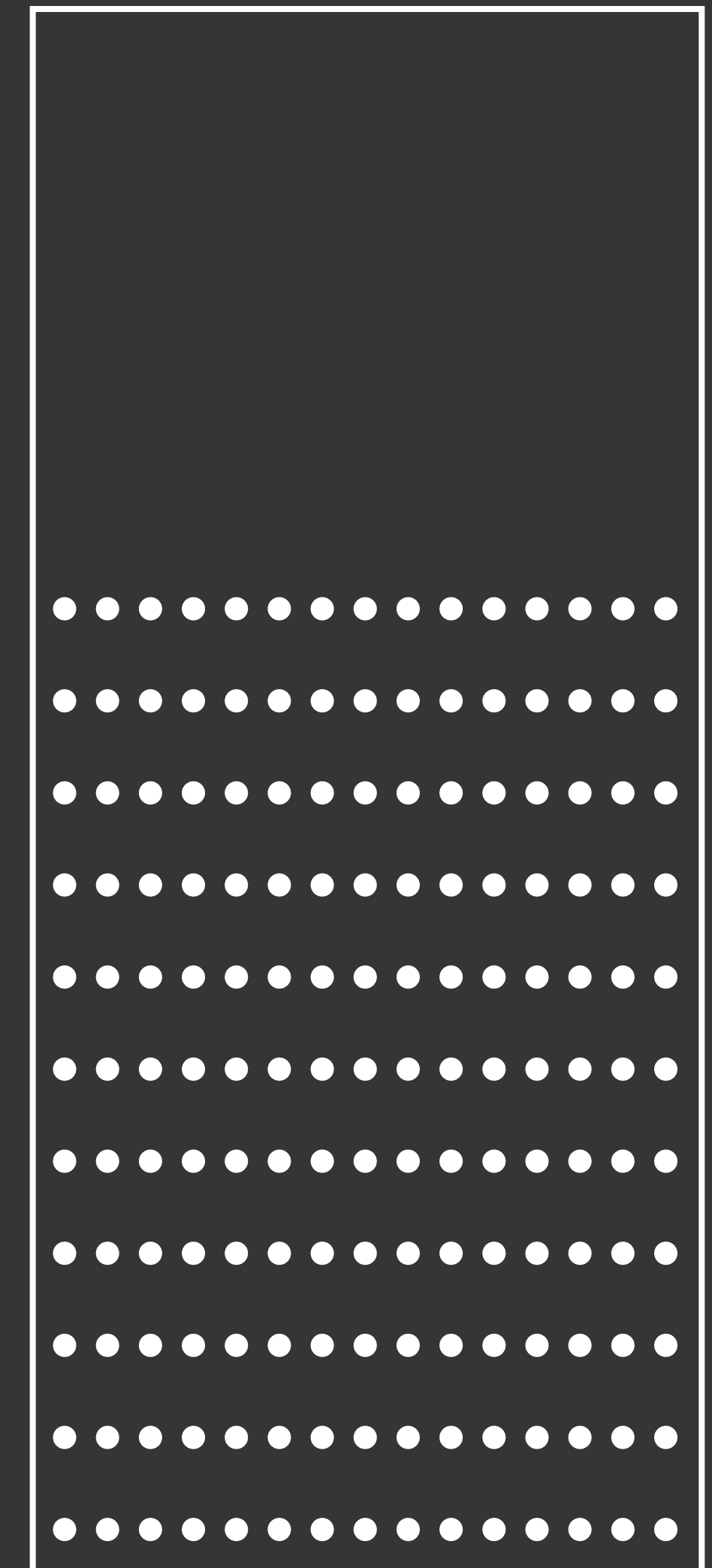
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4);
```



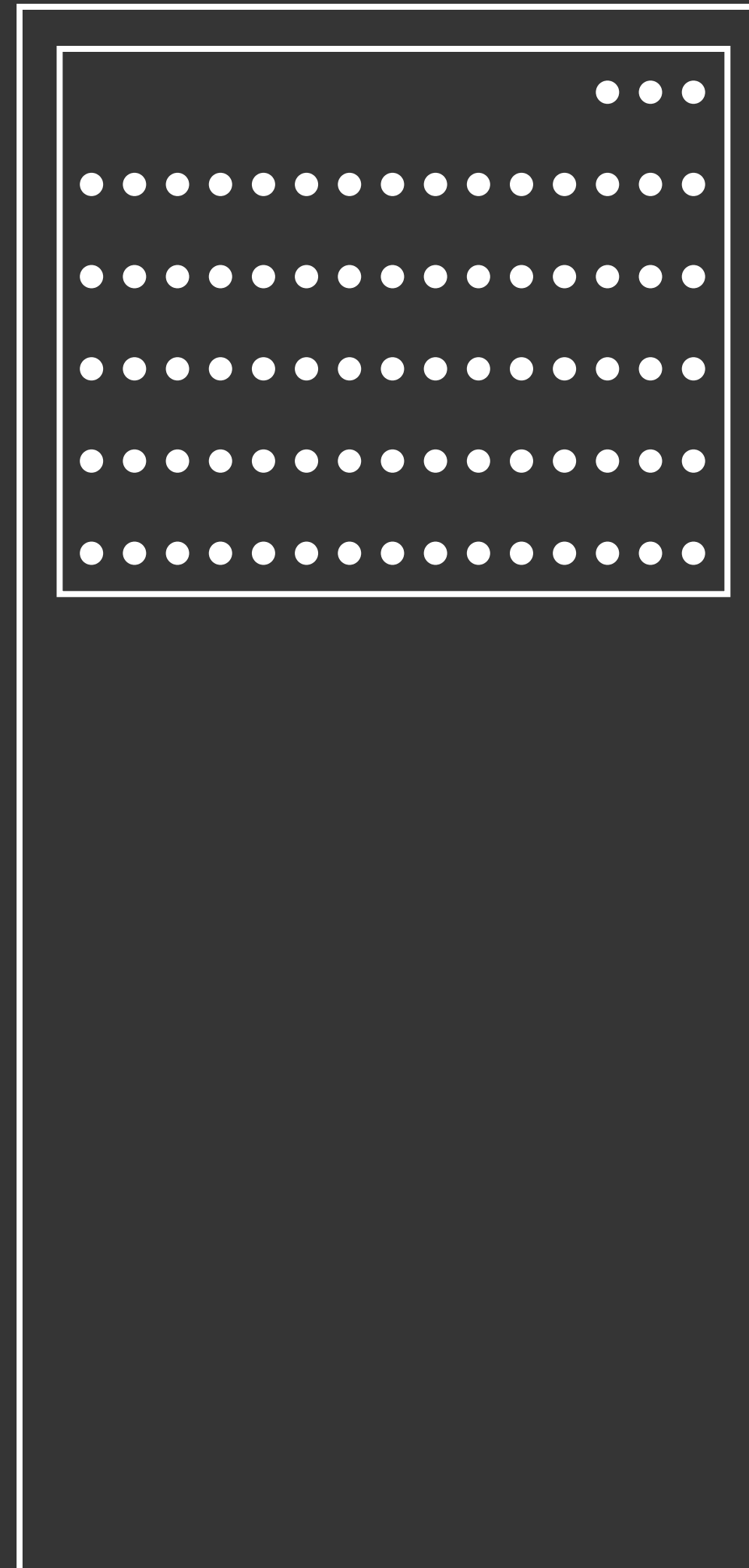
BufferedSource



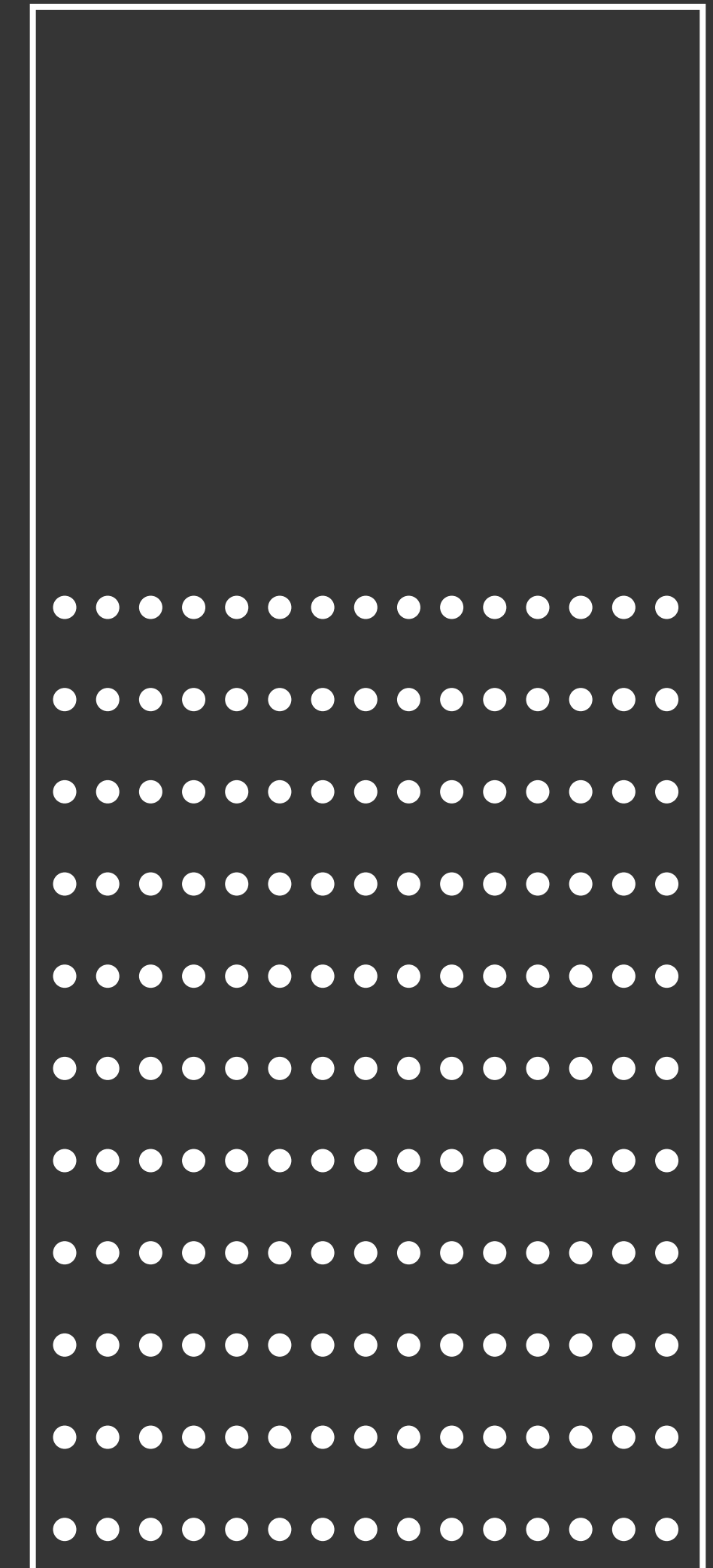
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....
```



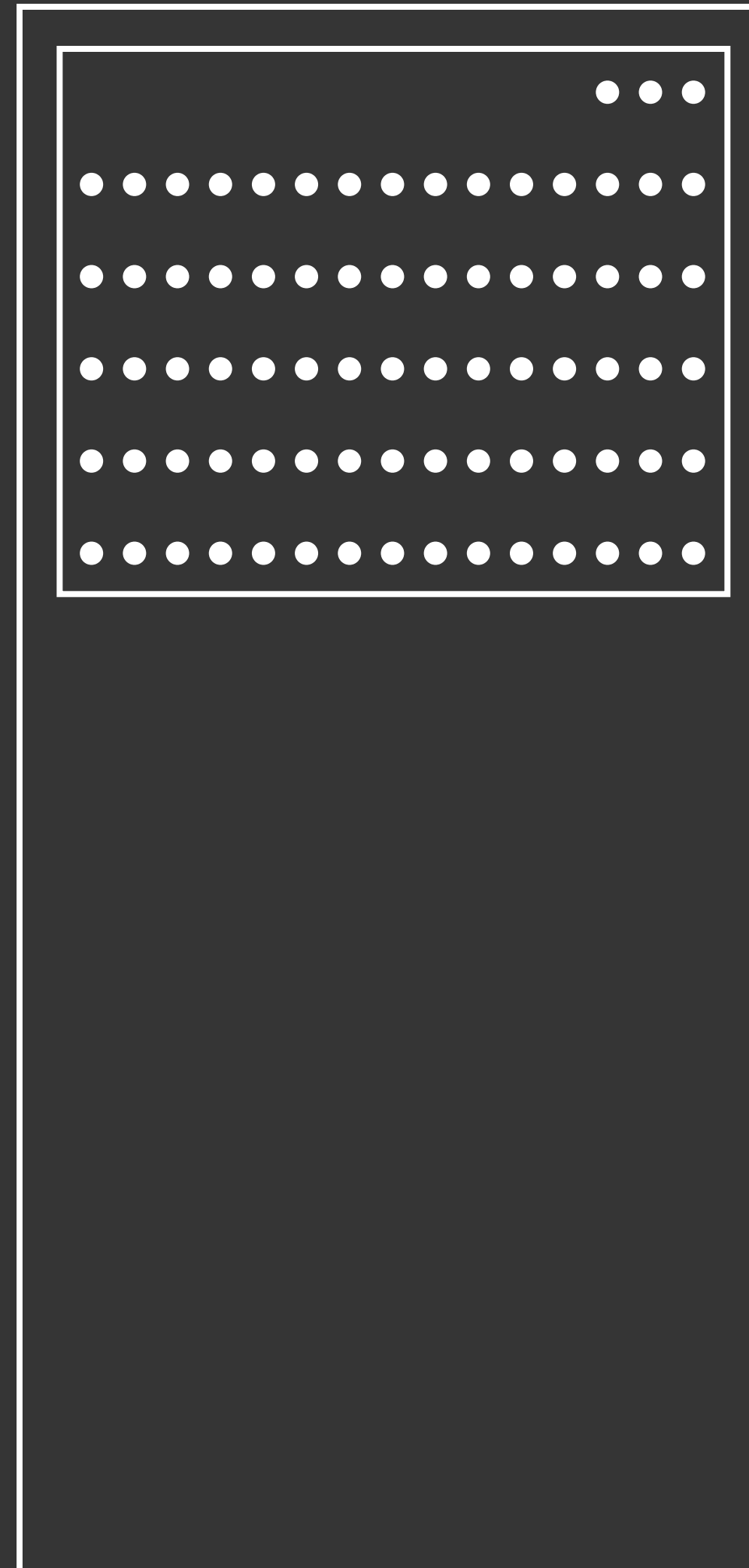
BufferedSource



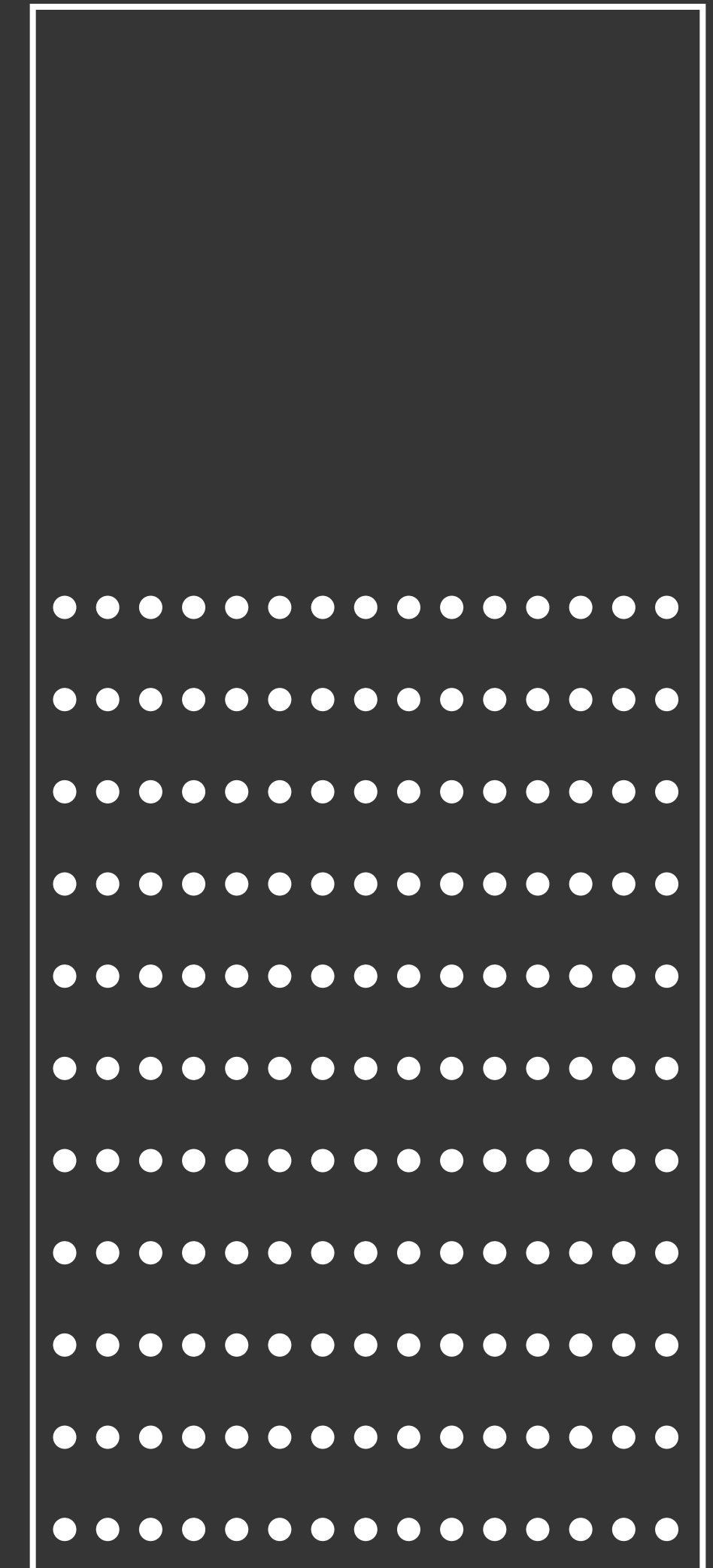
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong();
```



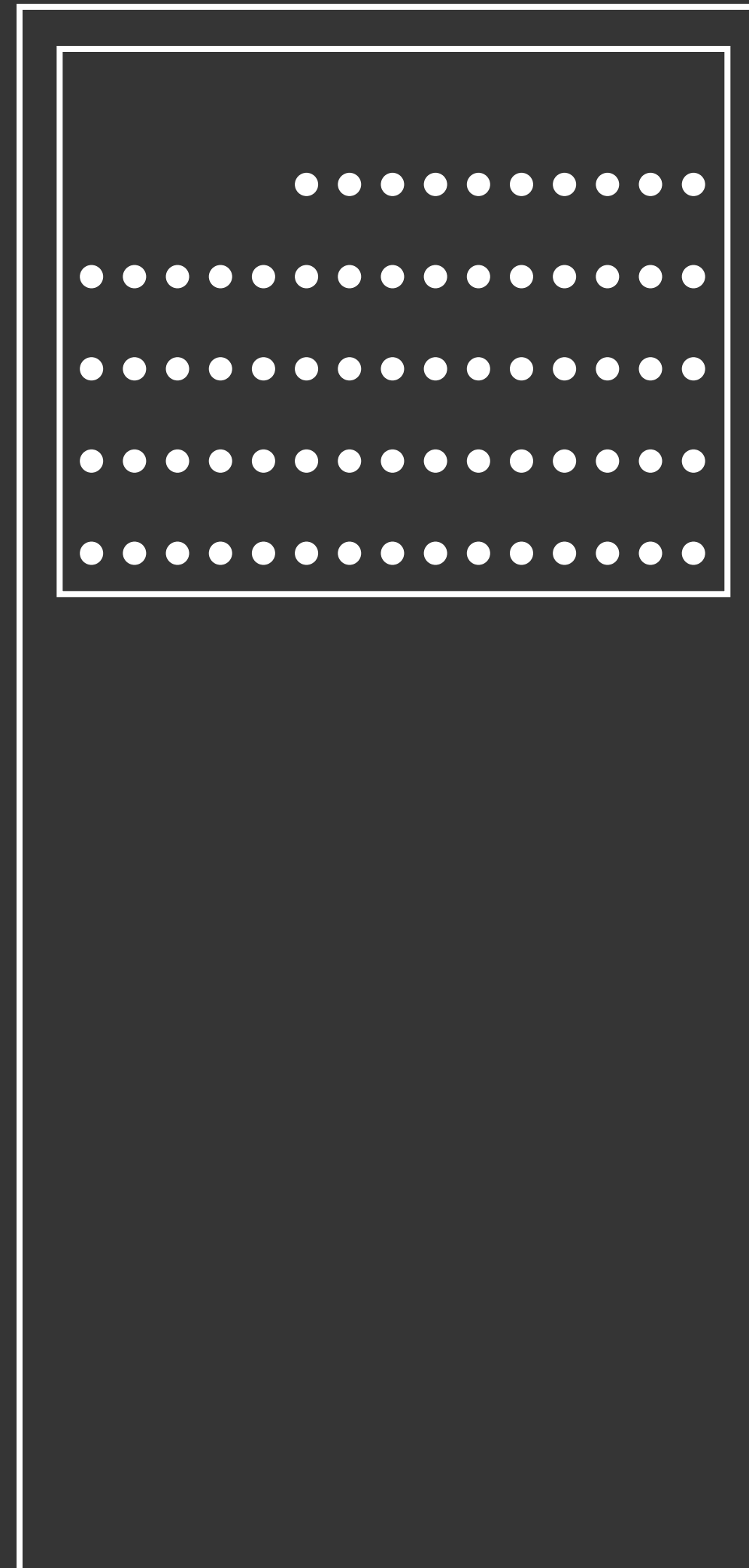
BufferedSource



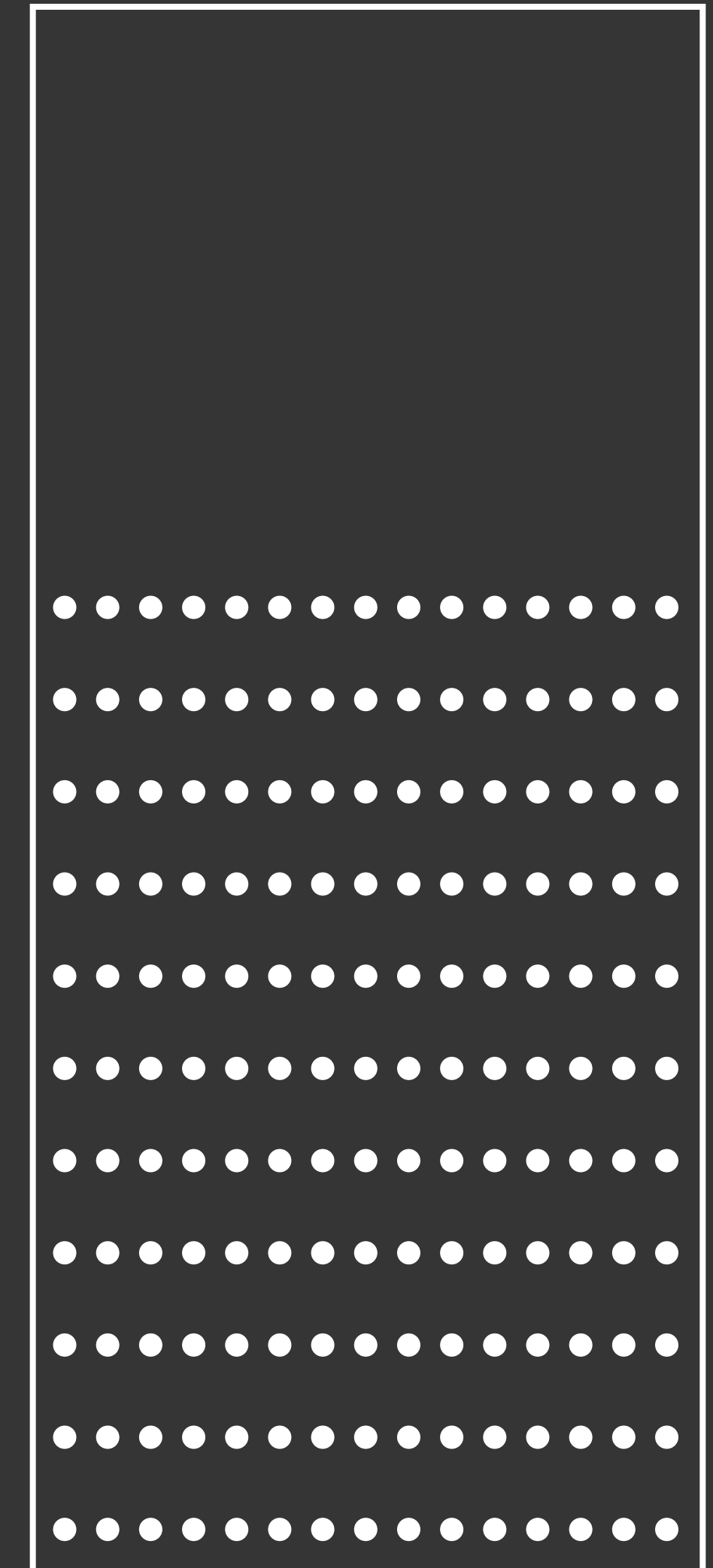
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....
```



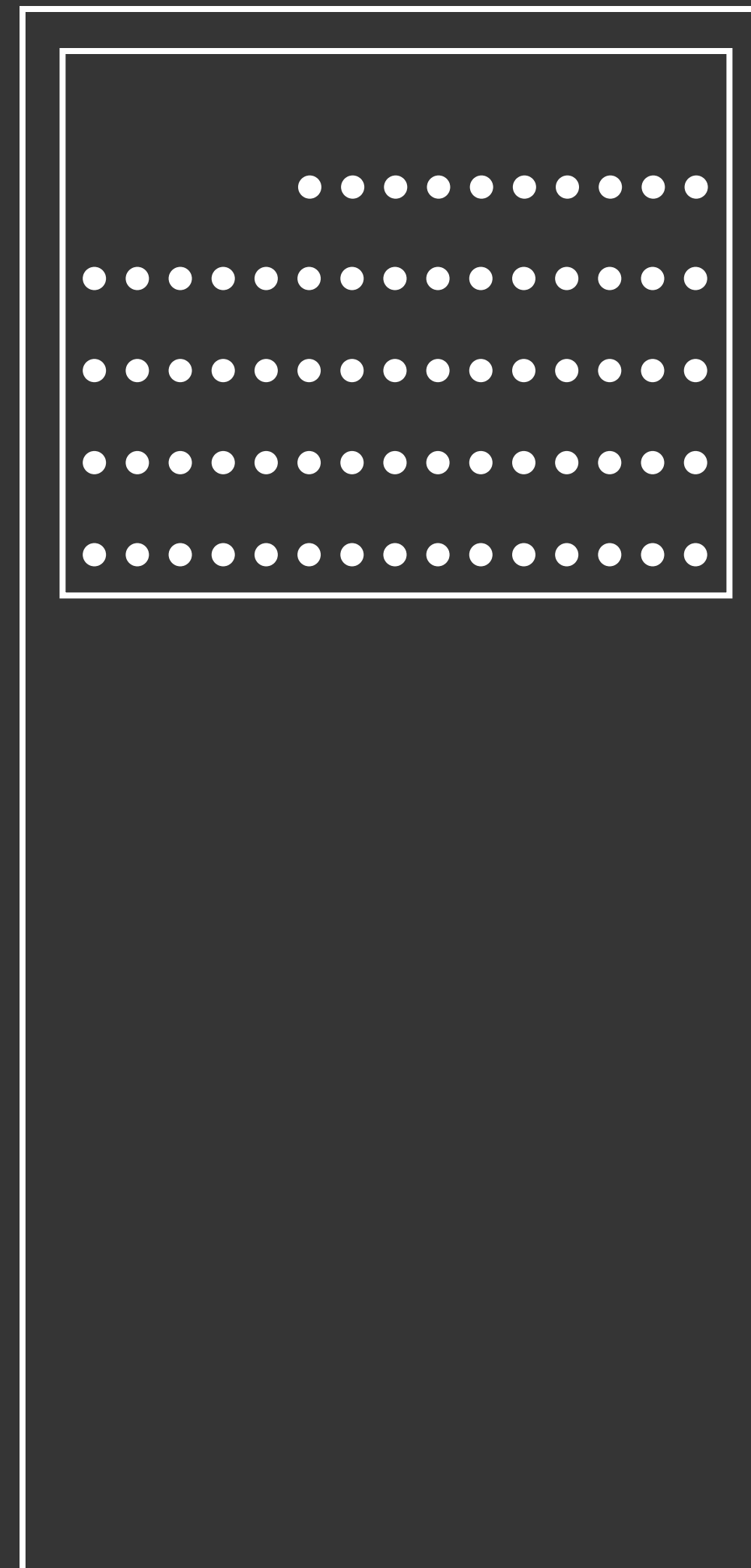
BufferedSource



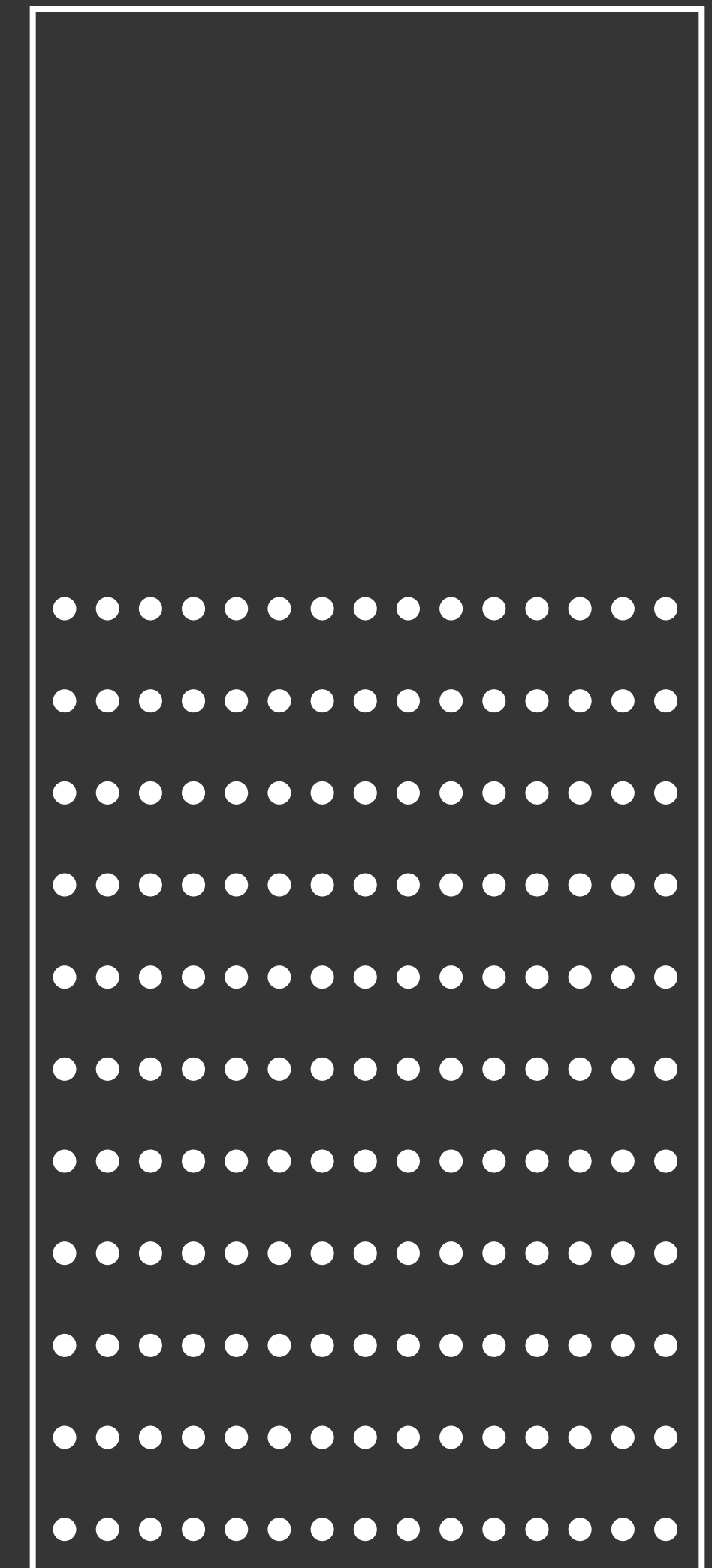
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();
```



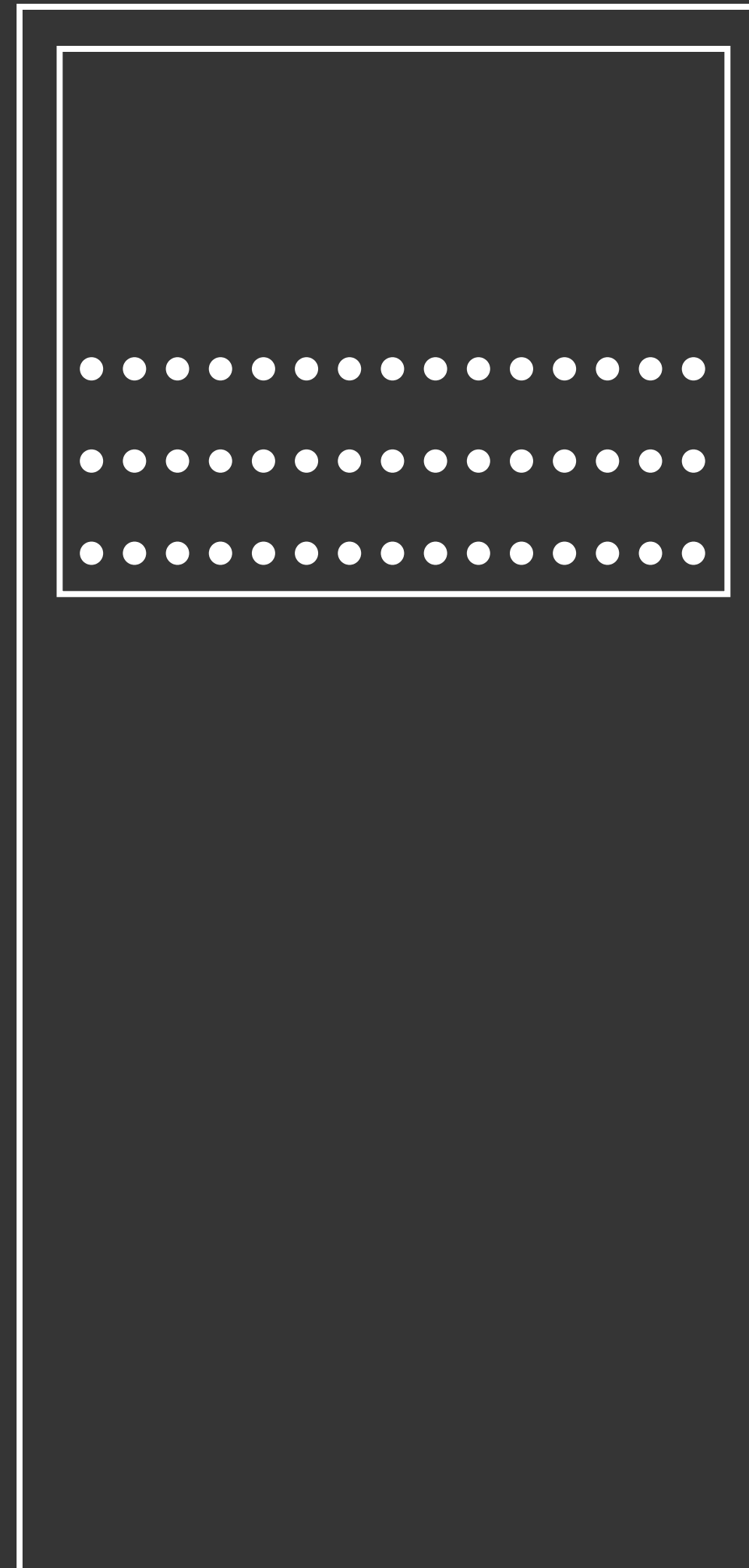
BufferedSource



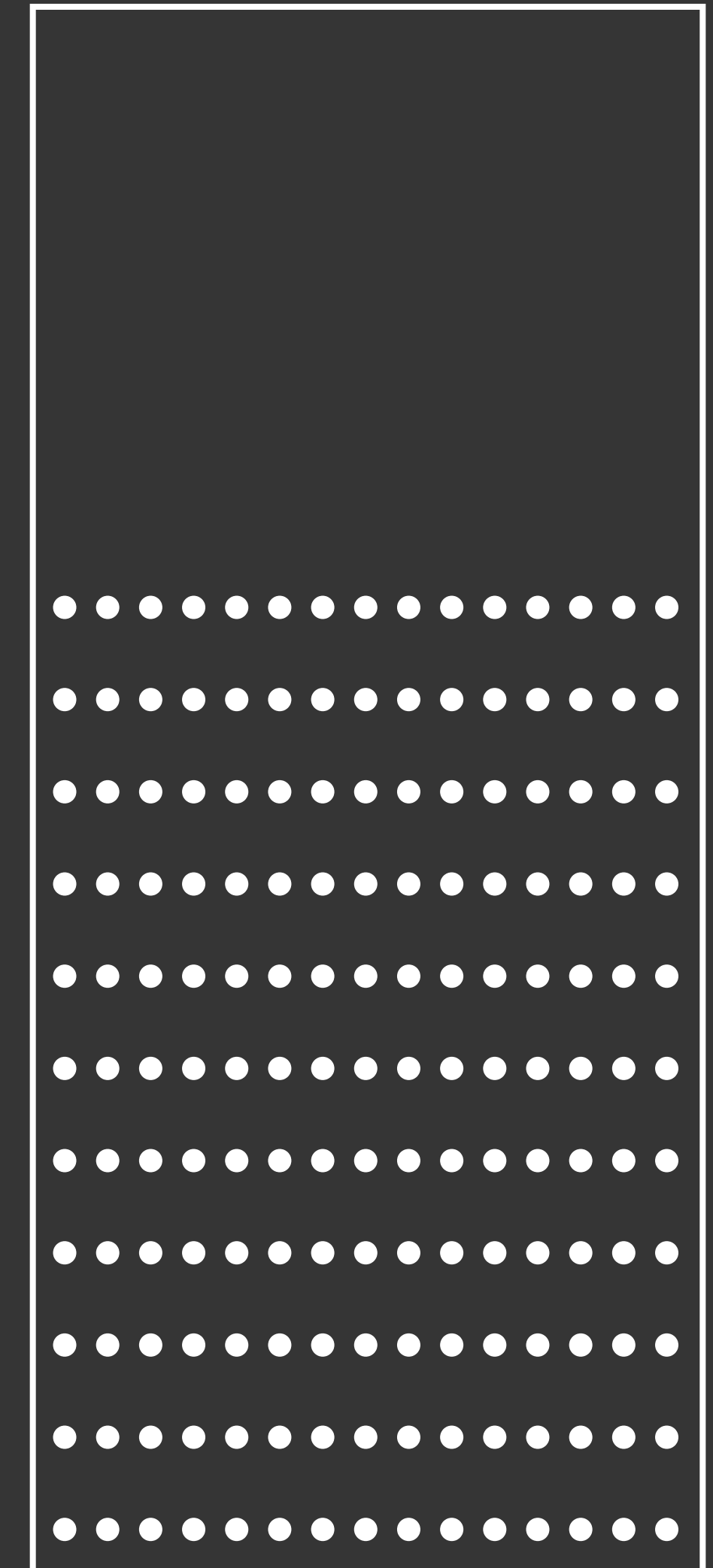
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();  
.....
```



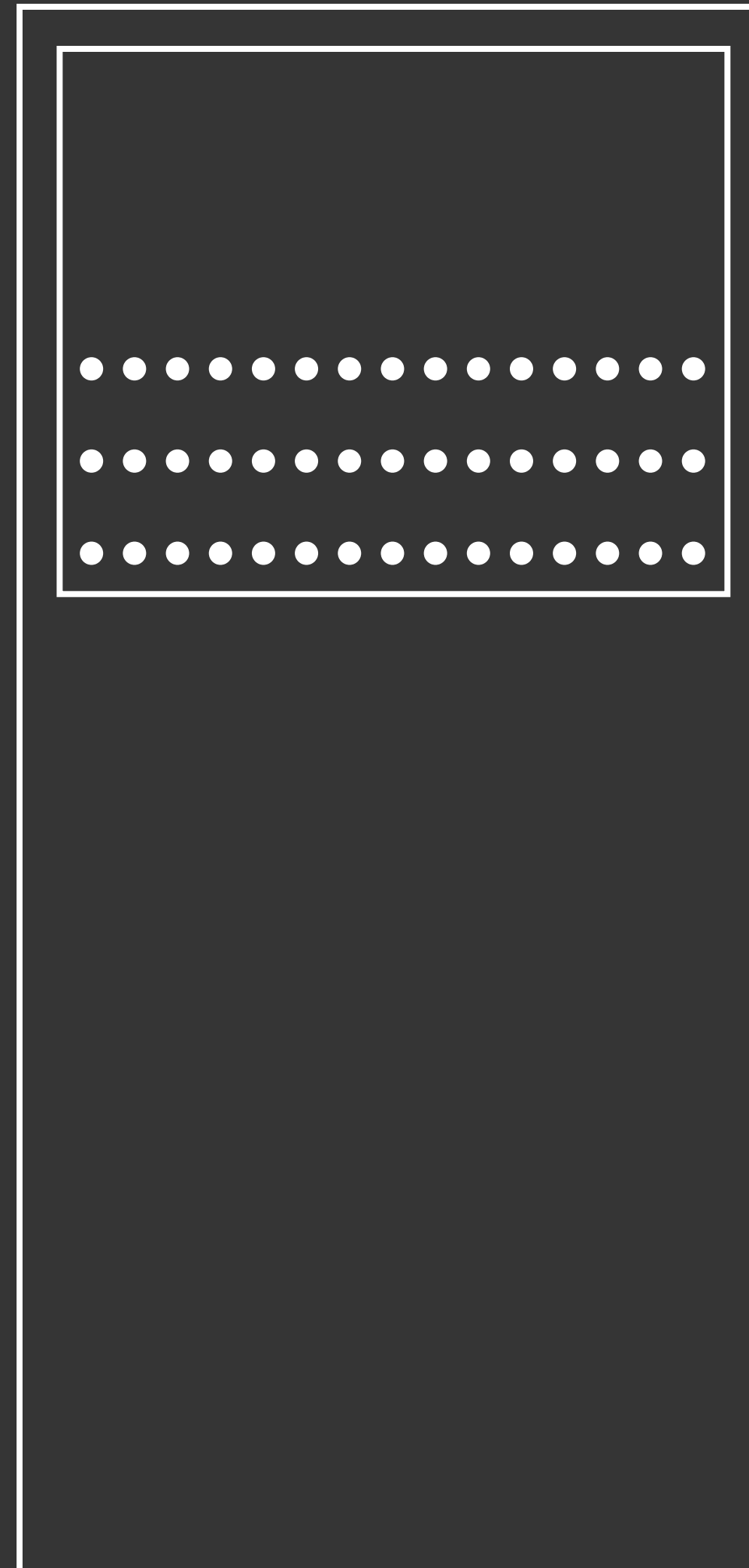
BufferedSource



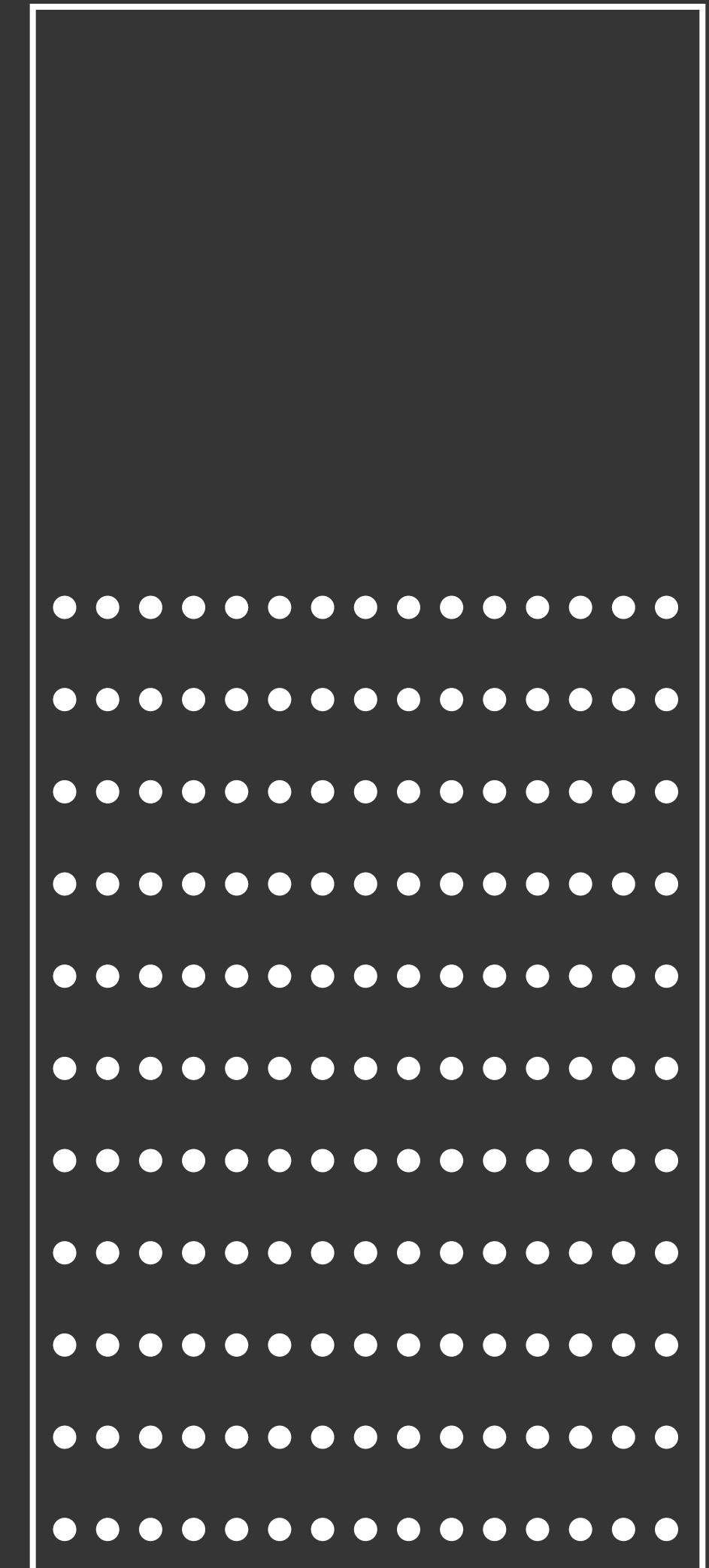
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();  
.....  
source.readUtf8LineStrict();
```



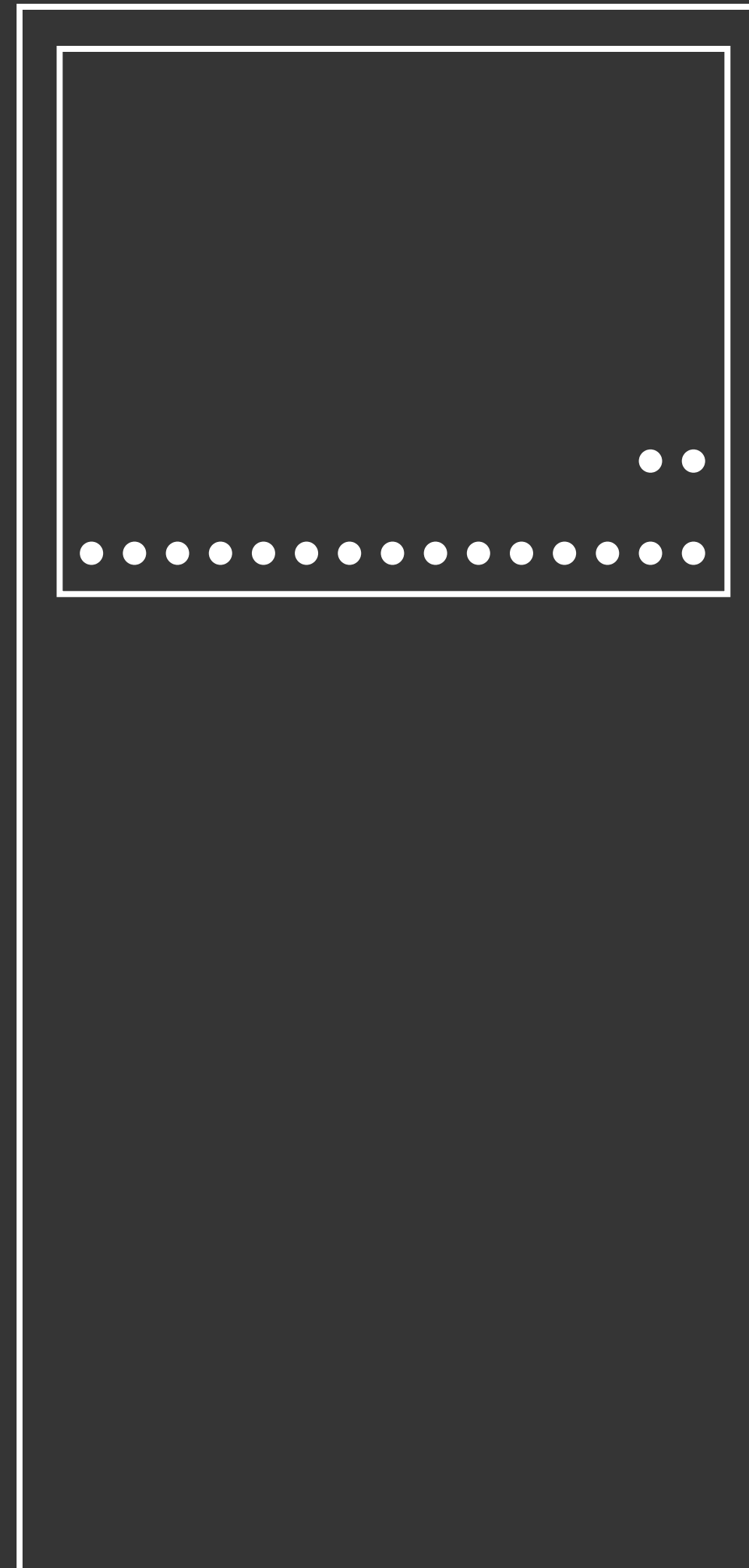
BufferedSource



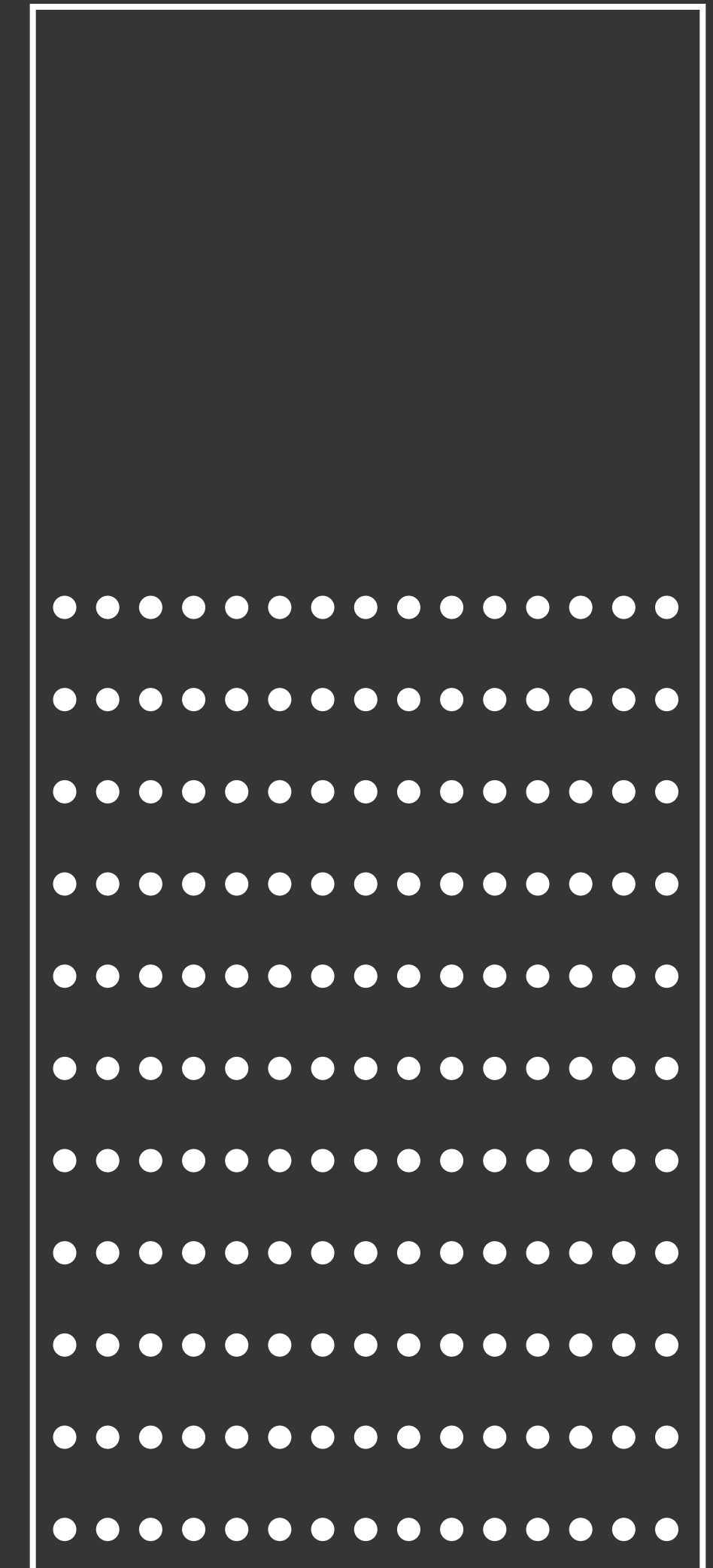
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();  
.....  
source.readUtf8LineStrict();  
.....
```



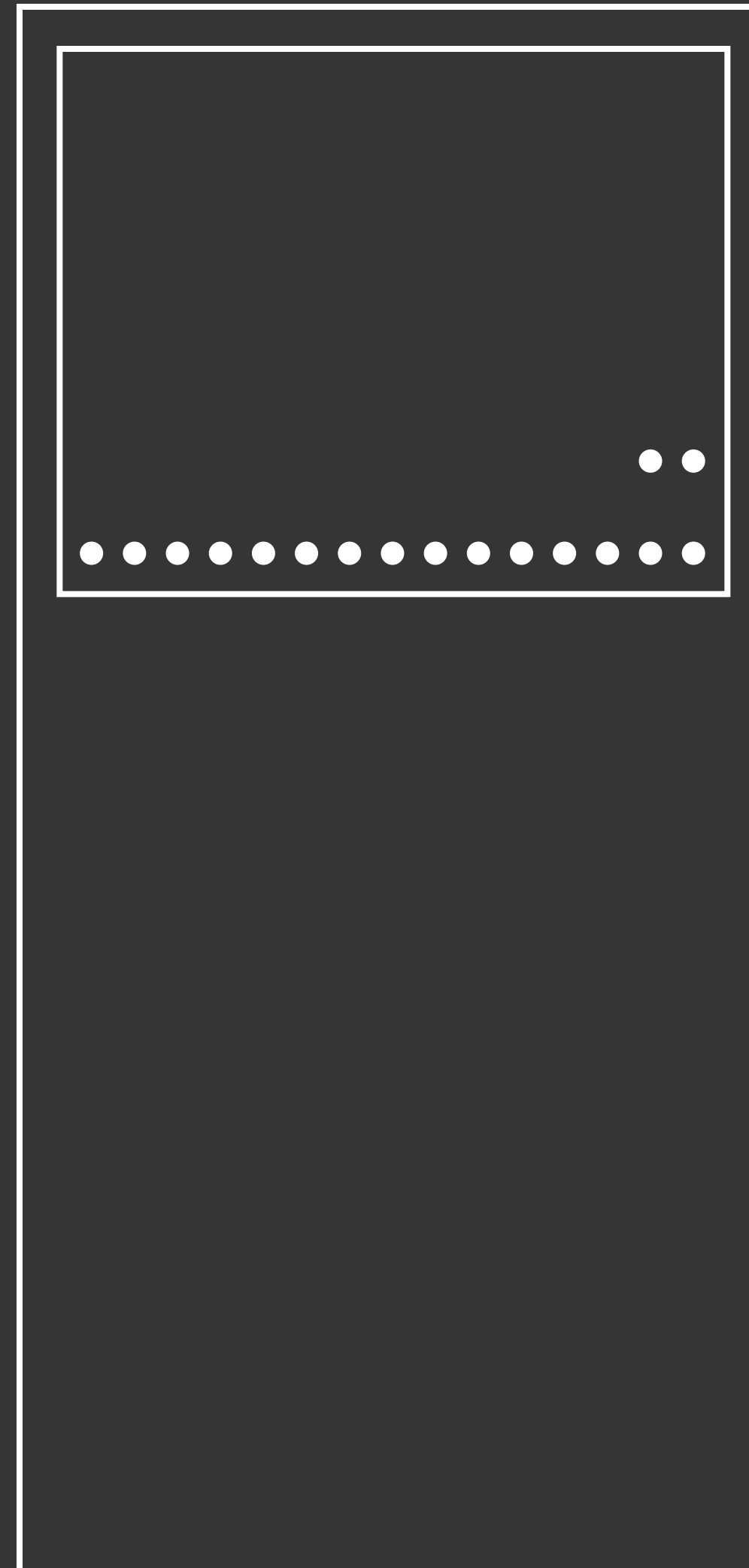
BufferedSource



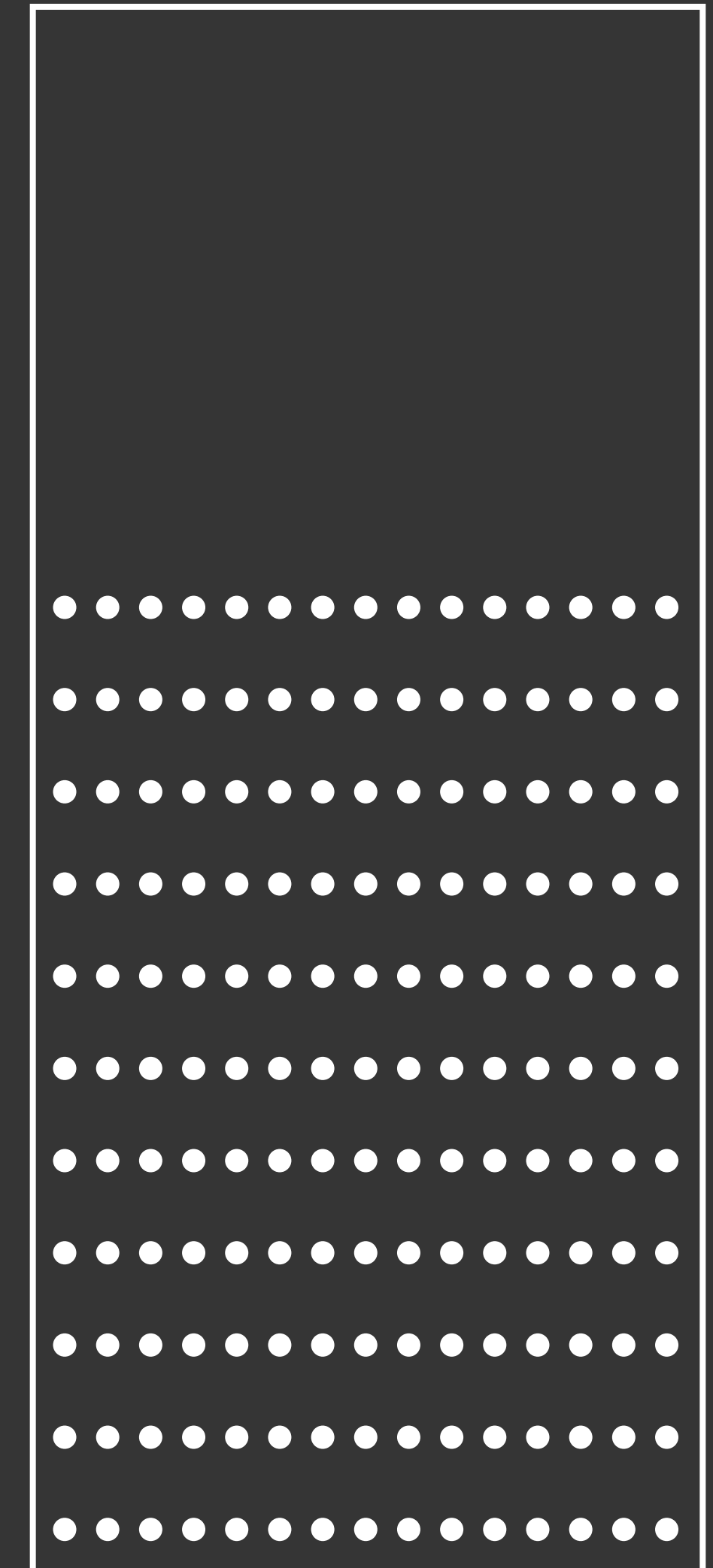
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();  
.....  
source.readUtf8LineStrict();  
.....  
source.readUtf8(40);
```



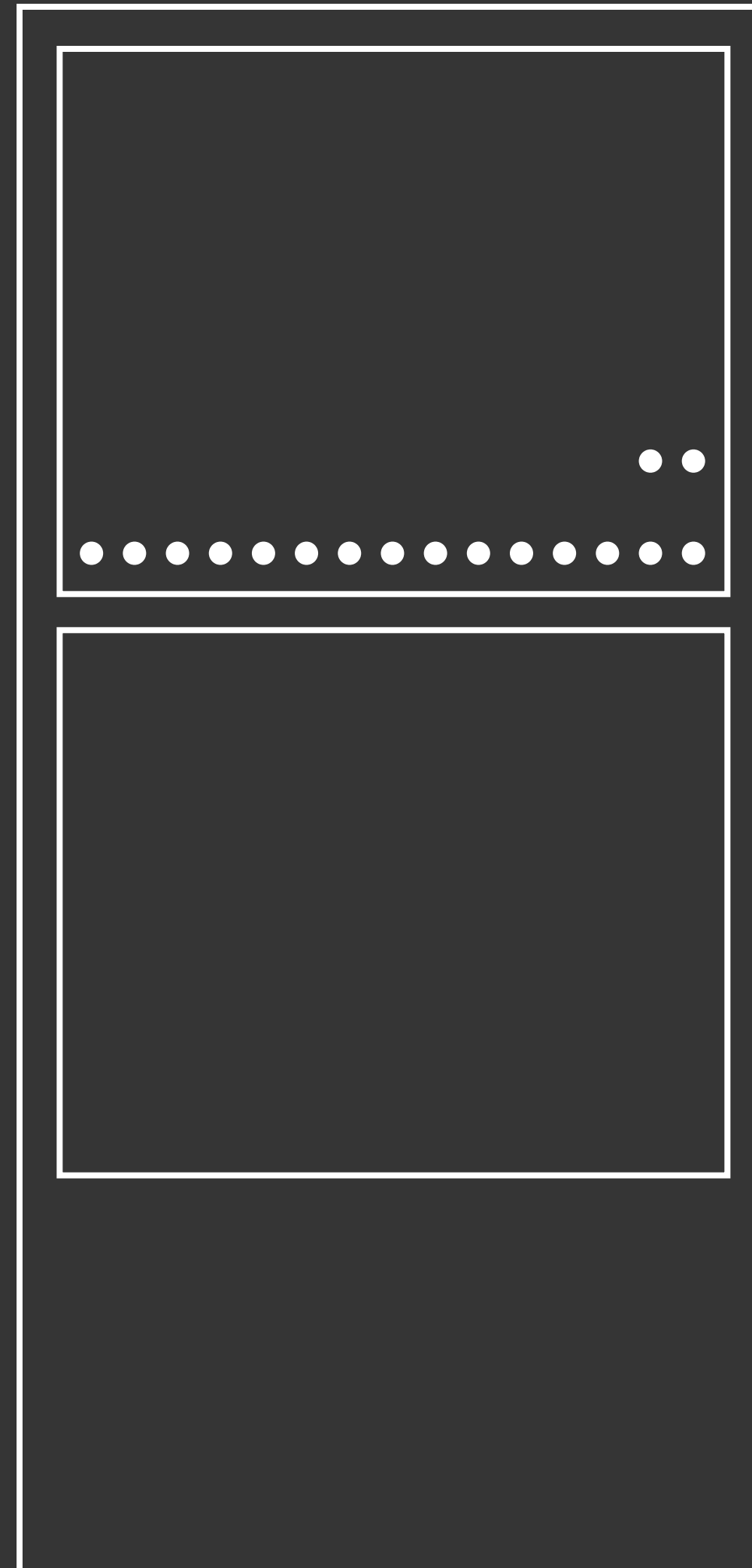
BufferedSource



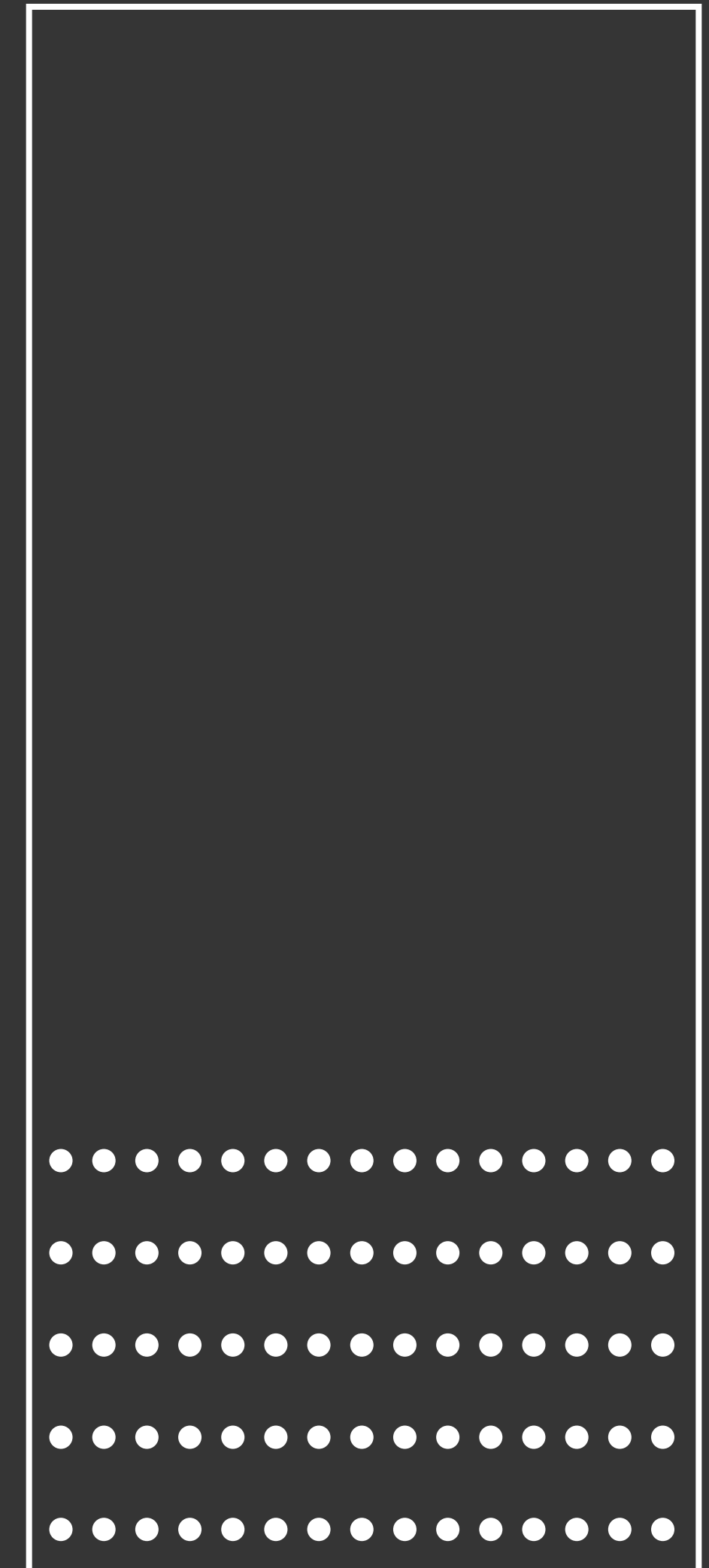
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();  
.....  
source.readUtf8LineStrict();  
.....  
source.readUtf8(40);
```



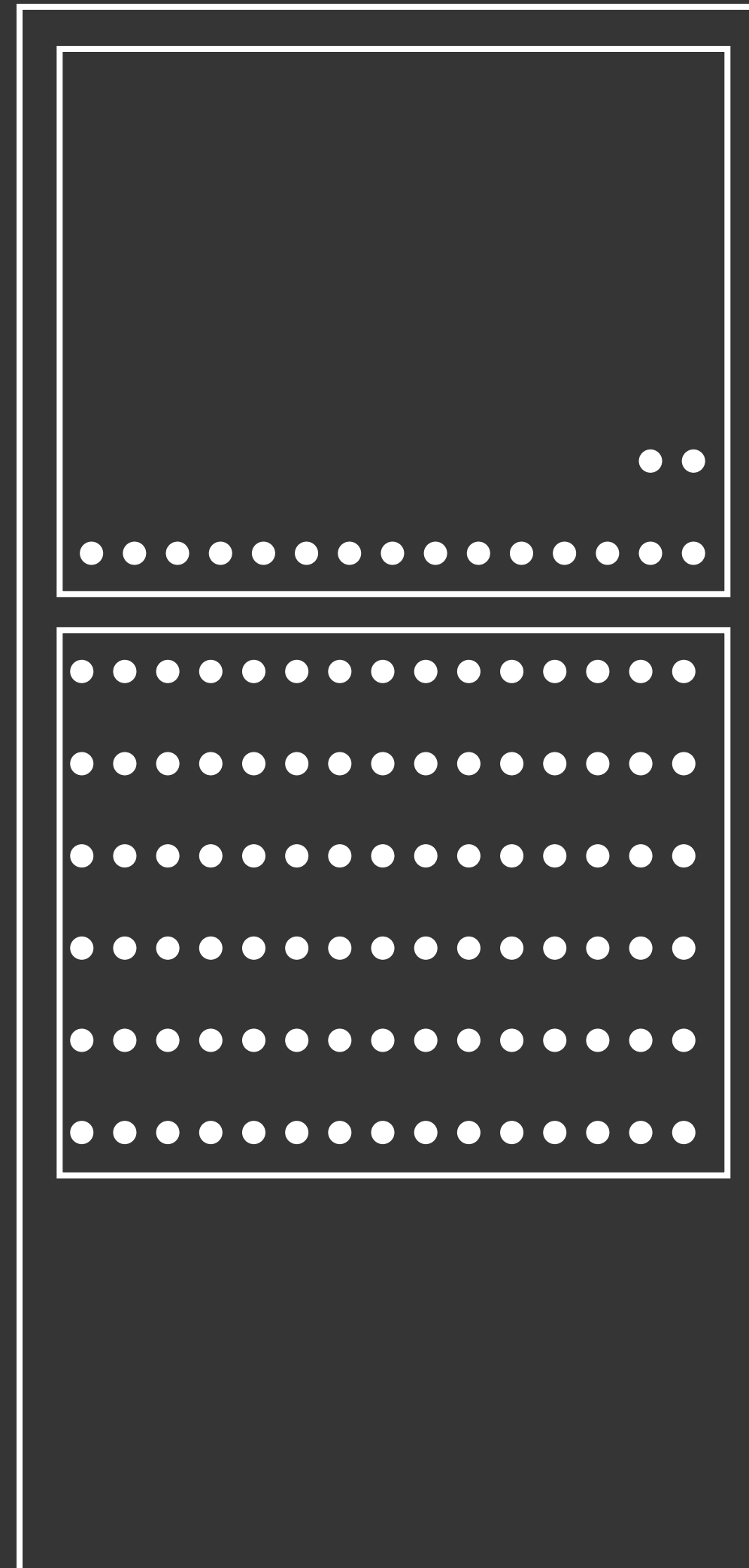
BufferedSource



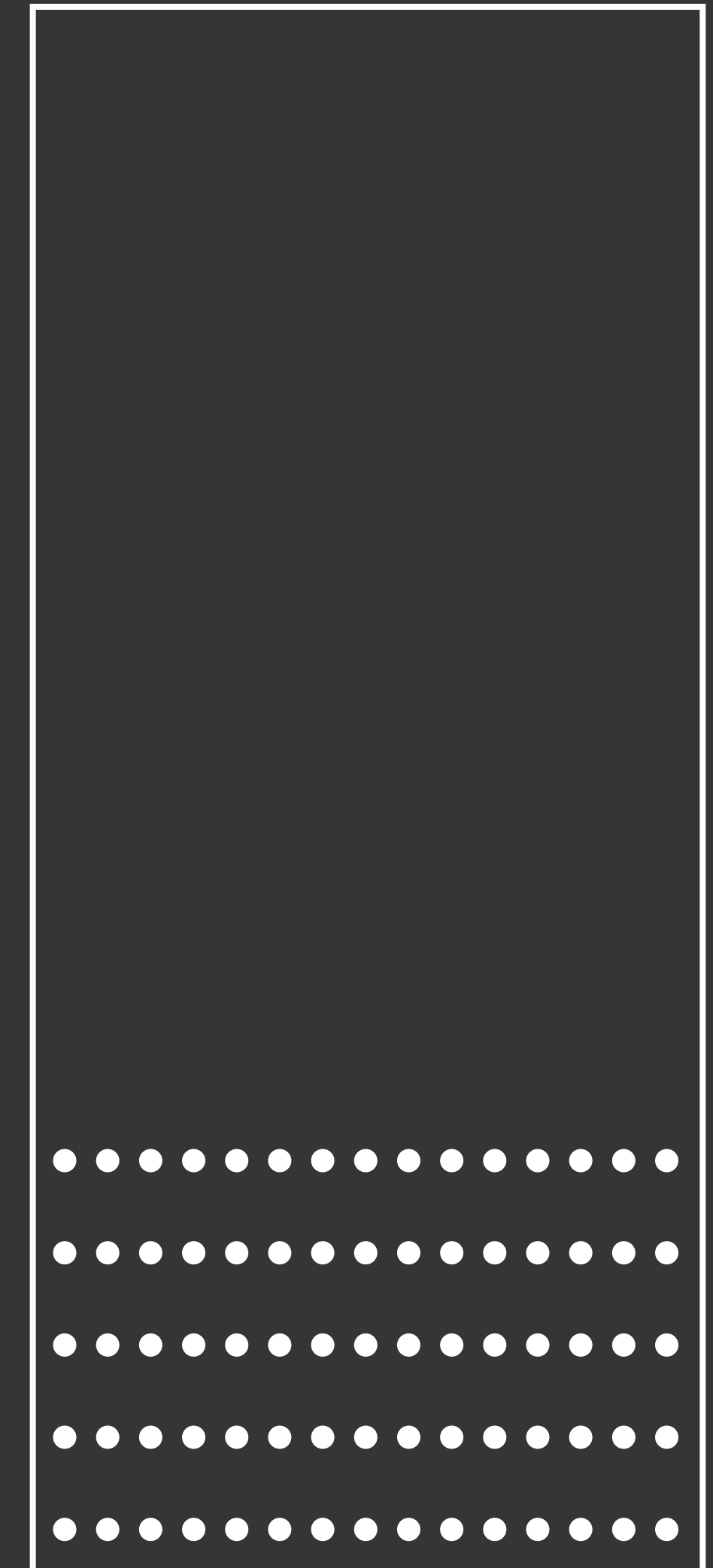
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();  
.....  
source.readUtf8LineStrict();  
.....  
source.readUtf8(40);
```



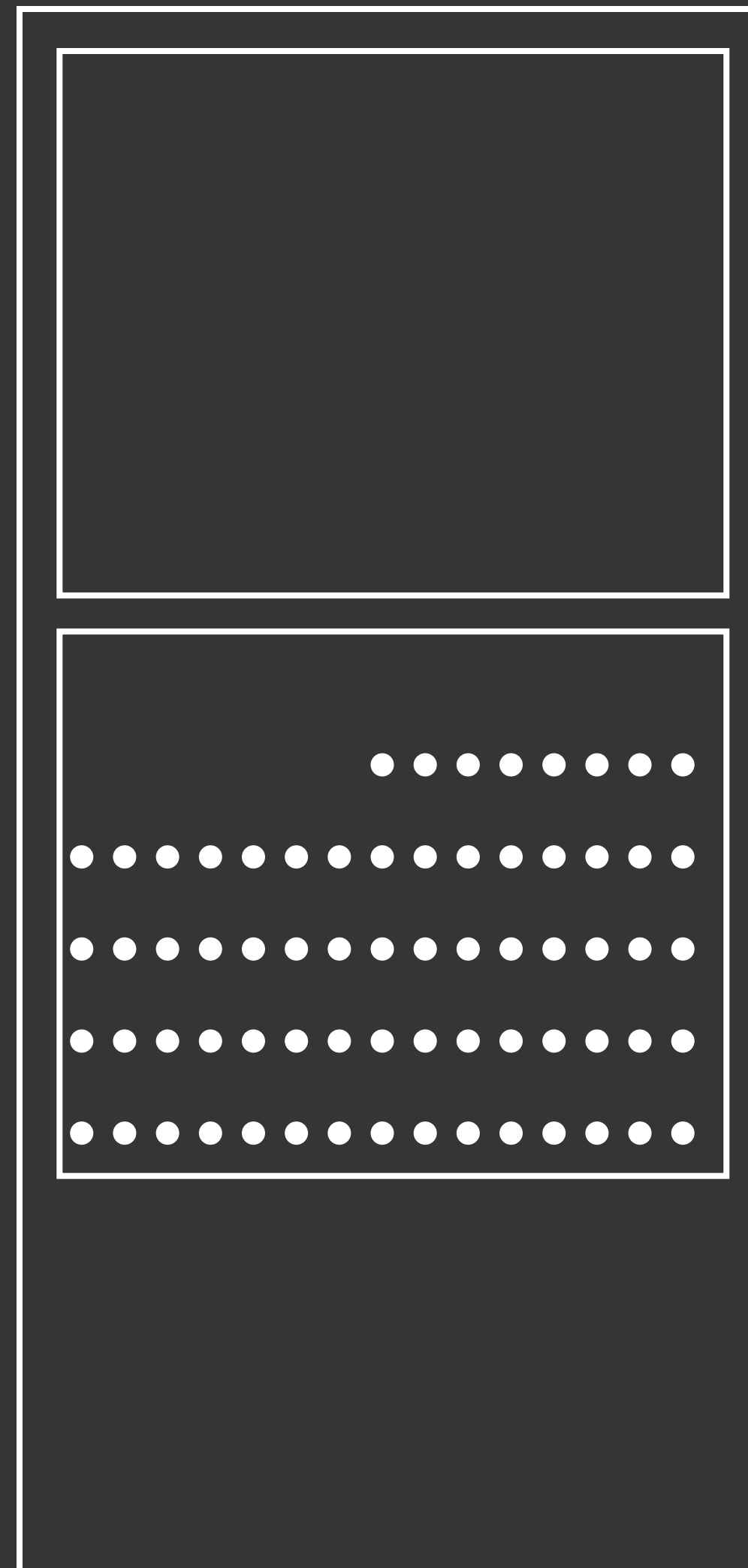
BufferedSource



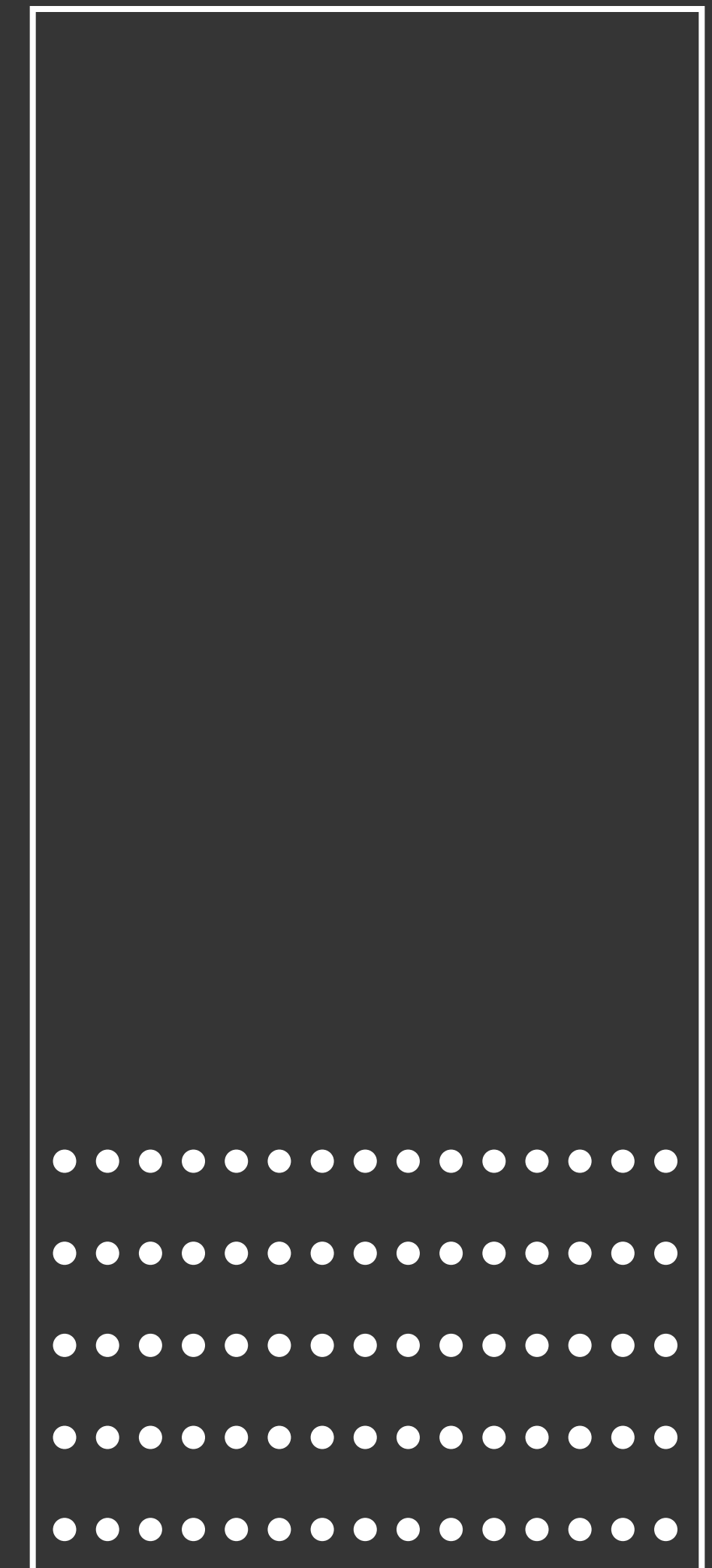
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();  
.....  
source.readUtf8LineStrict();  
.....  
source.readUtf8(40);  
.....  
.....
```



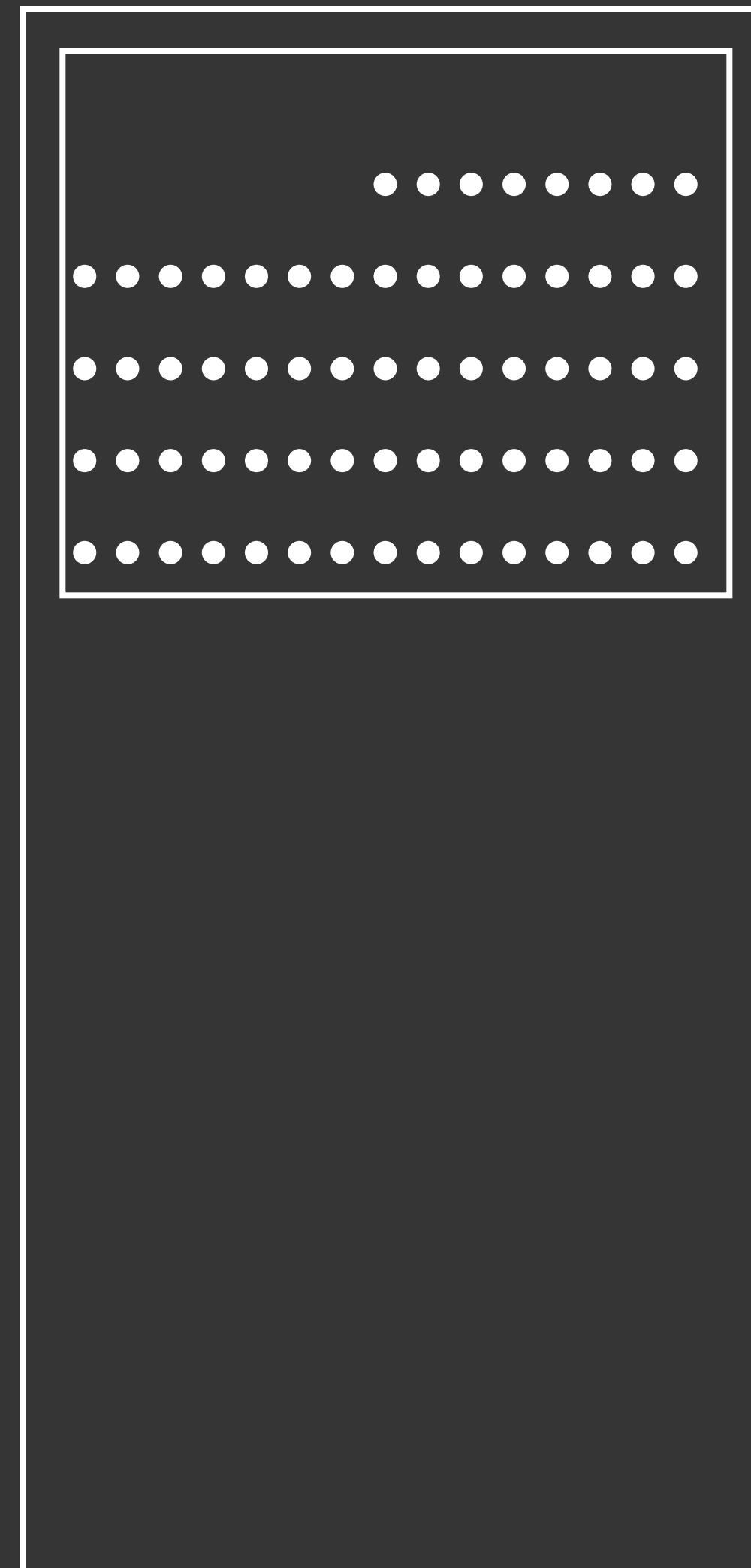
BufferedSource



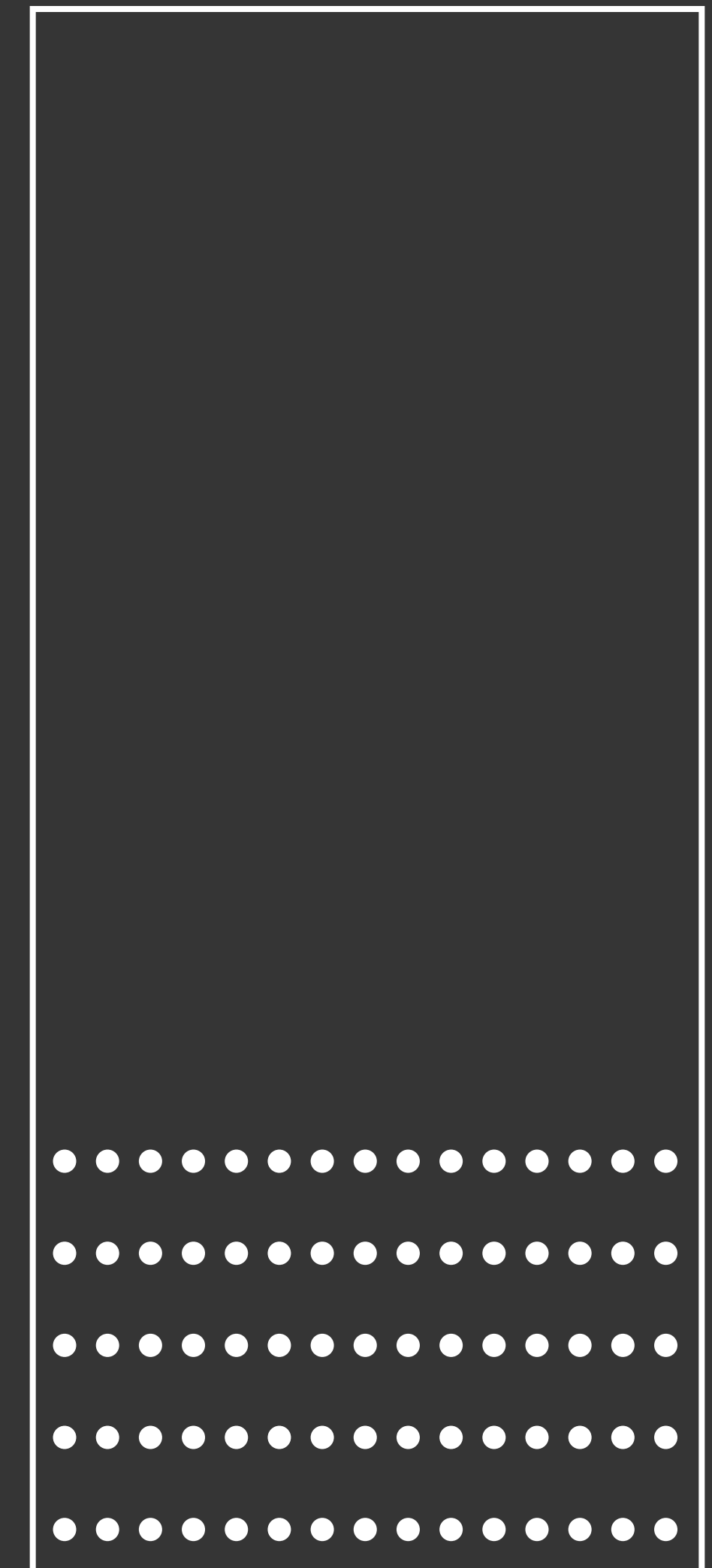
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();  
.....  
source.readUtf8LineStrict();  
.....  
source.readUtf8(40);  
.....  
.....
```



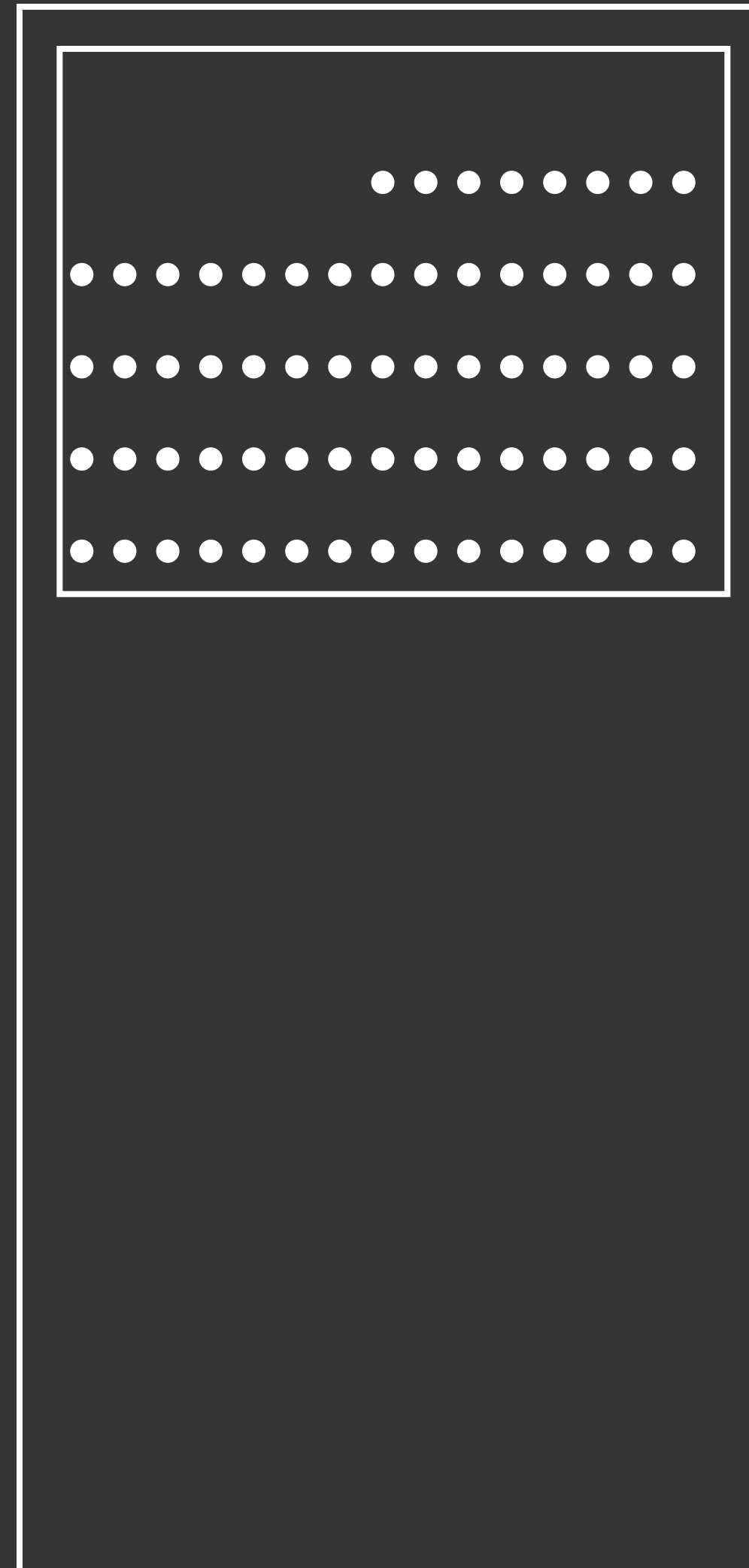
BufferedSource



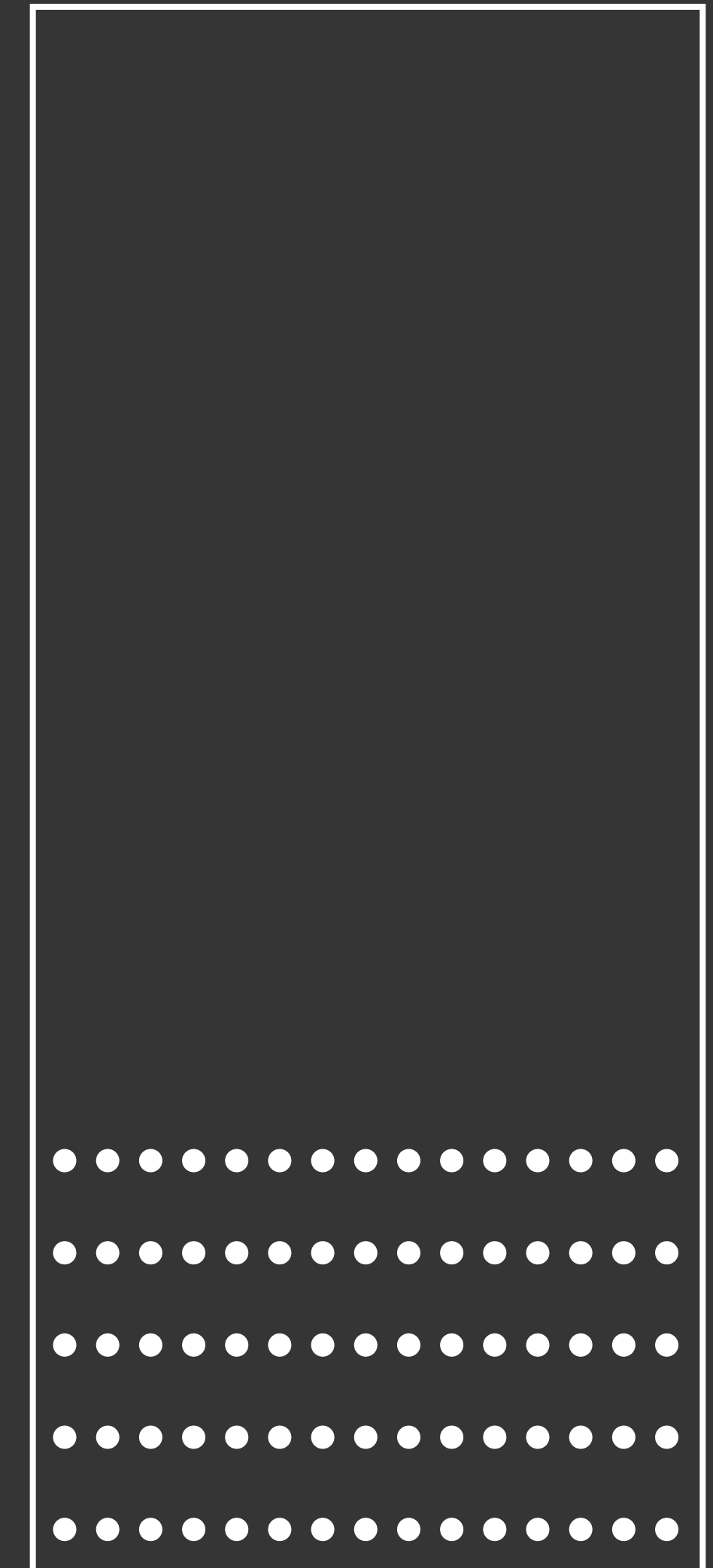
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();  
.....  
source.readUtf8LineStrict();  
.....  
source.readUtf8(40);  
.....  
.....  
source.readUtf8(40);
```



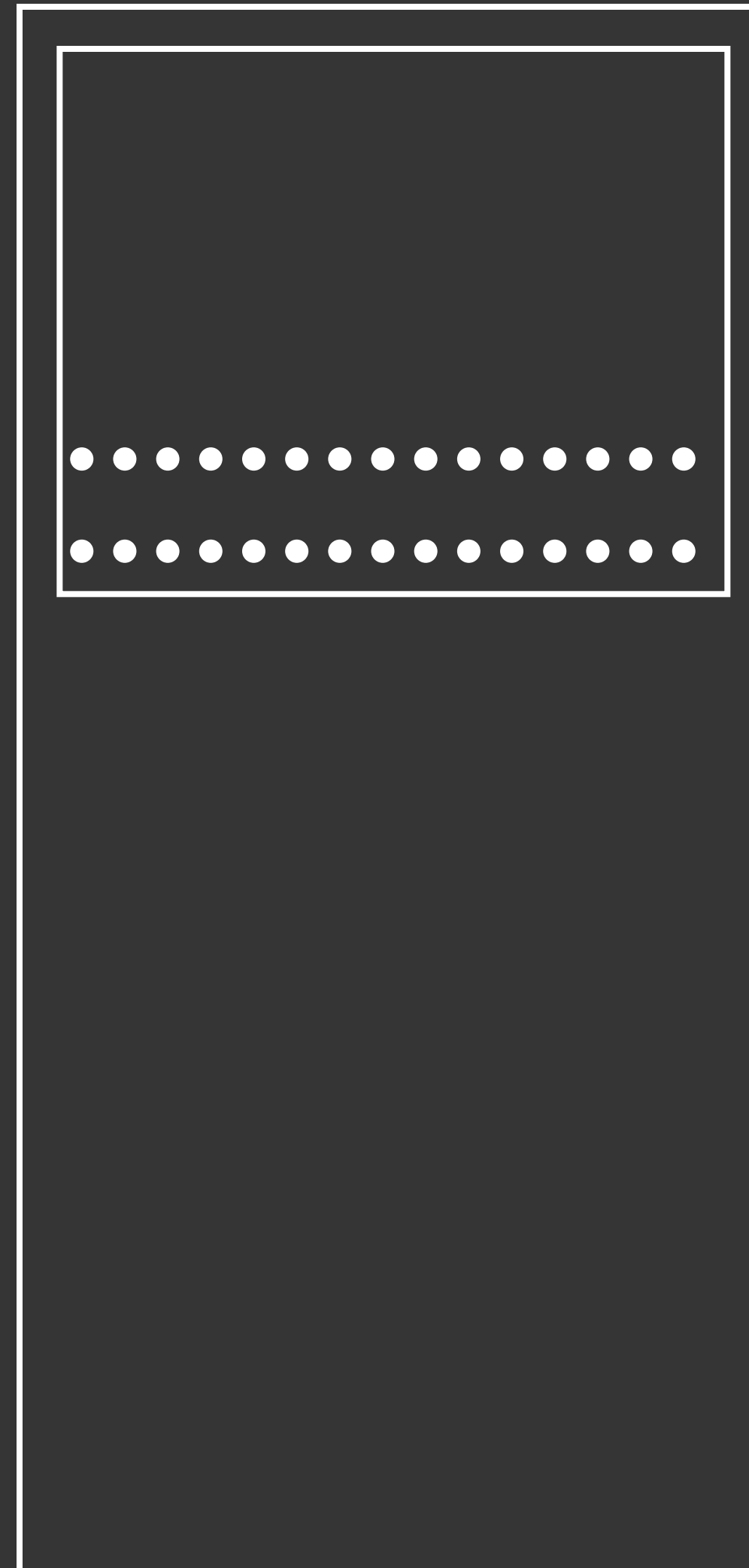
BufferedSource



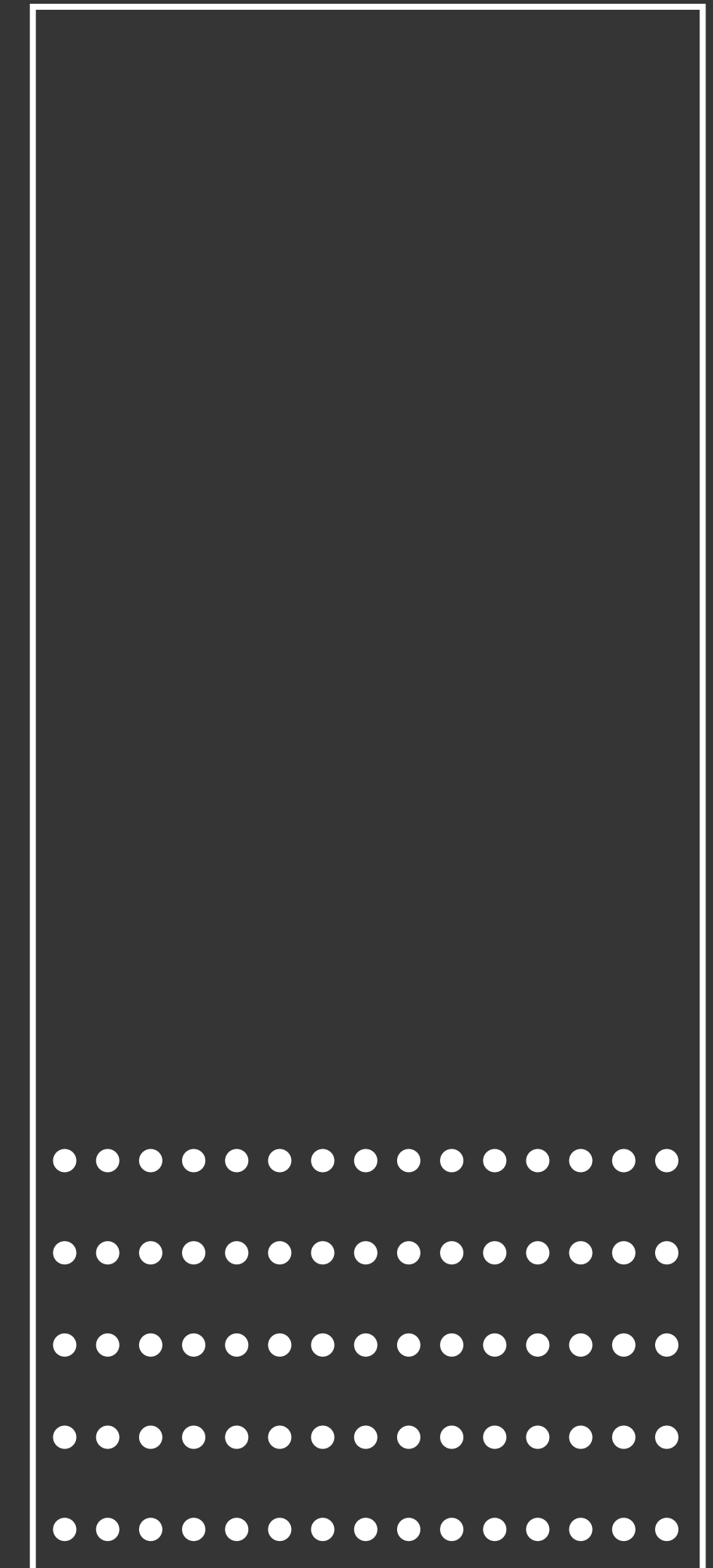
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();  
.....  
source.readUtf8LineStrict();  
.....  
source.readUtf8(40);  
.....  
.....  
source.readUtf8(40);  
.....  
.....
```



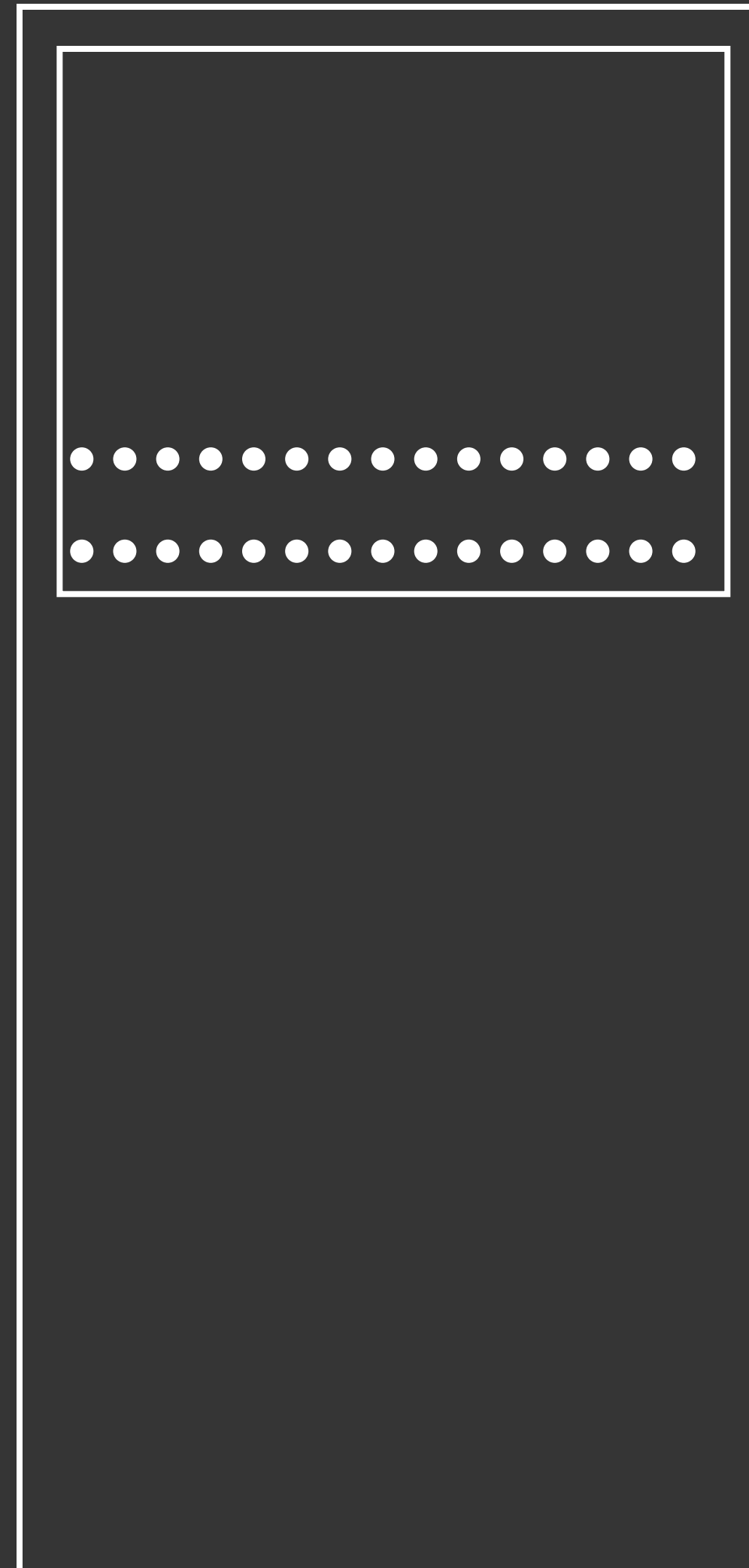
BufferedSource



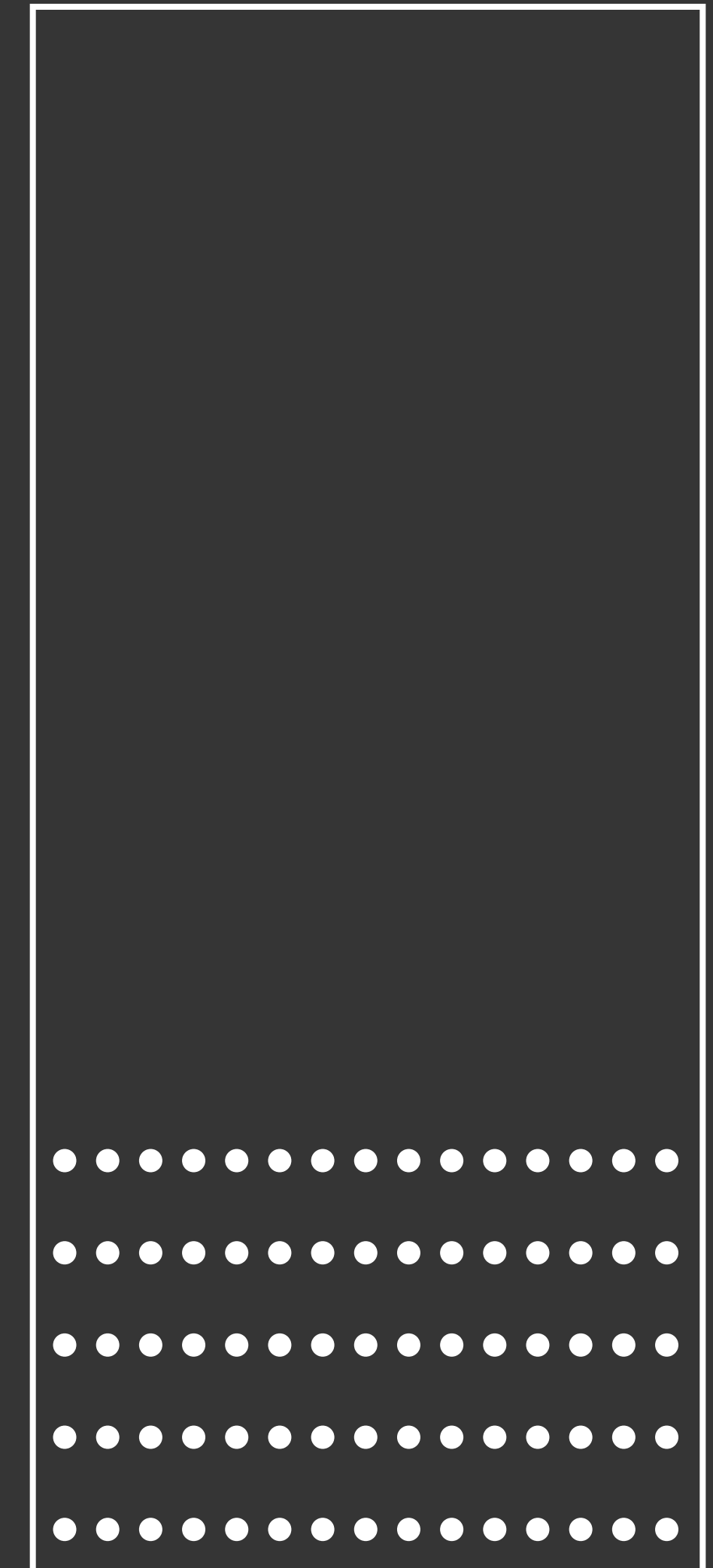
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();  
.....  
source.readUtf8LineStrict();  
.....  
source.readUtf8(40);  
.....  
.....  
source.readUtf8(40);  
.....  
.....  
source.readInt();
```



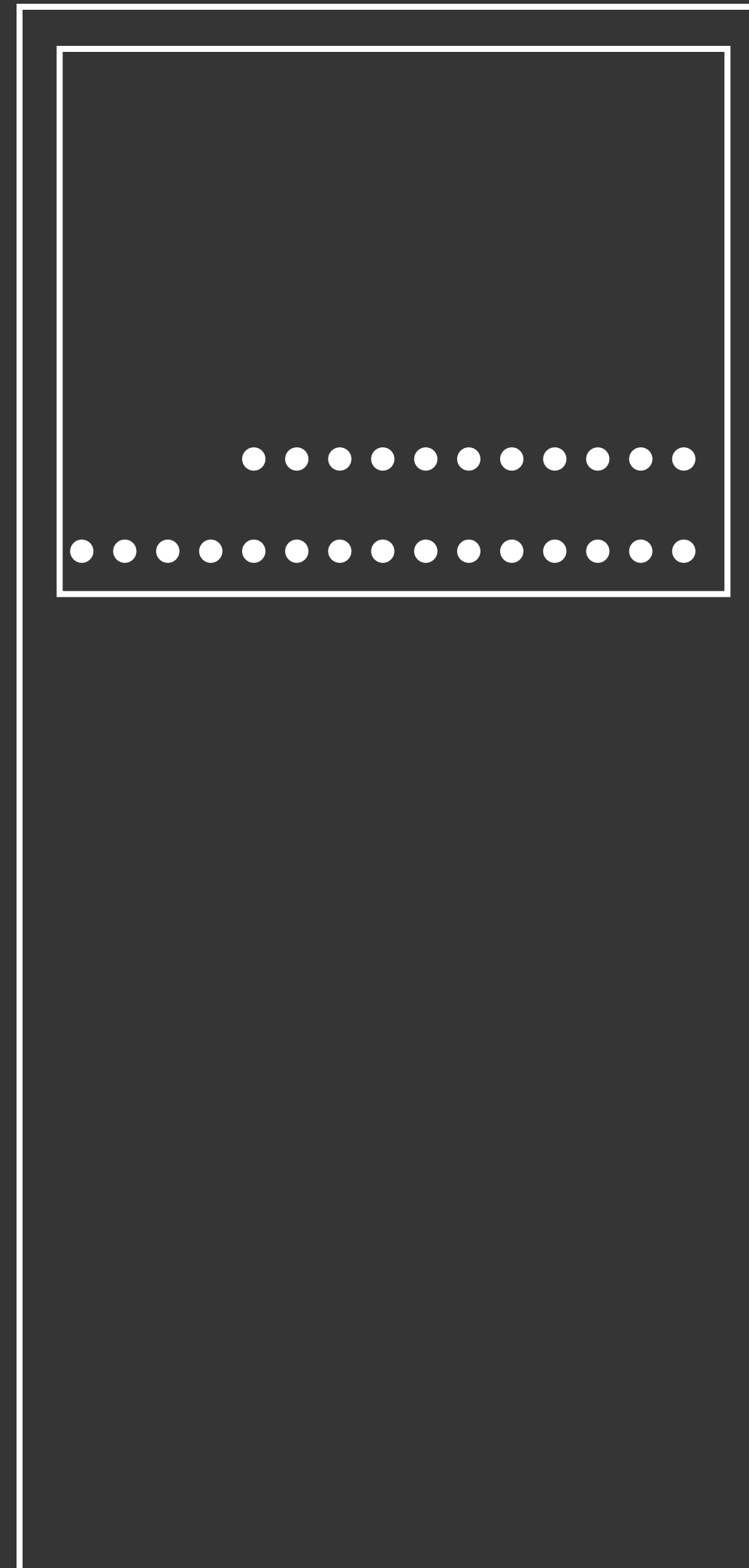
BufferedSource



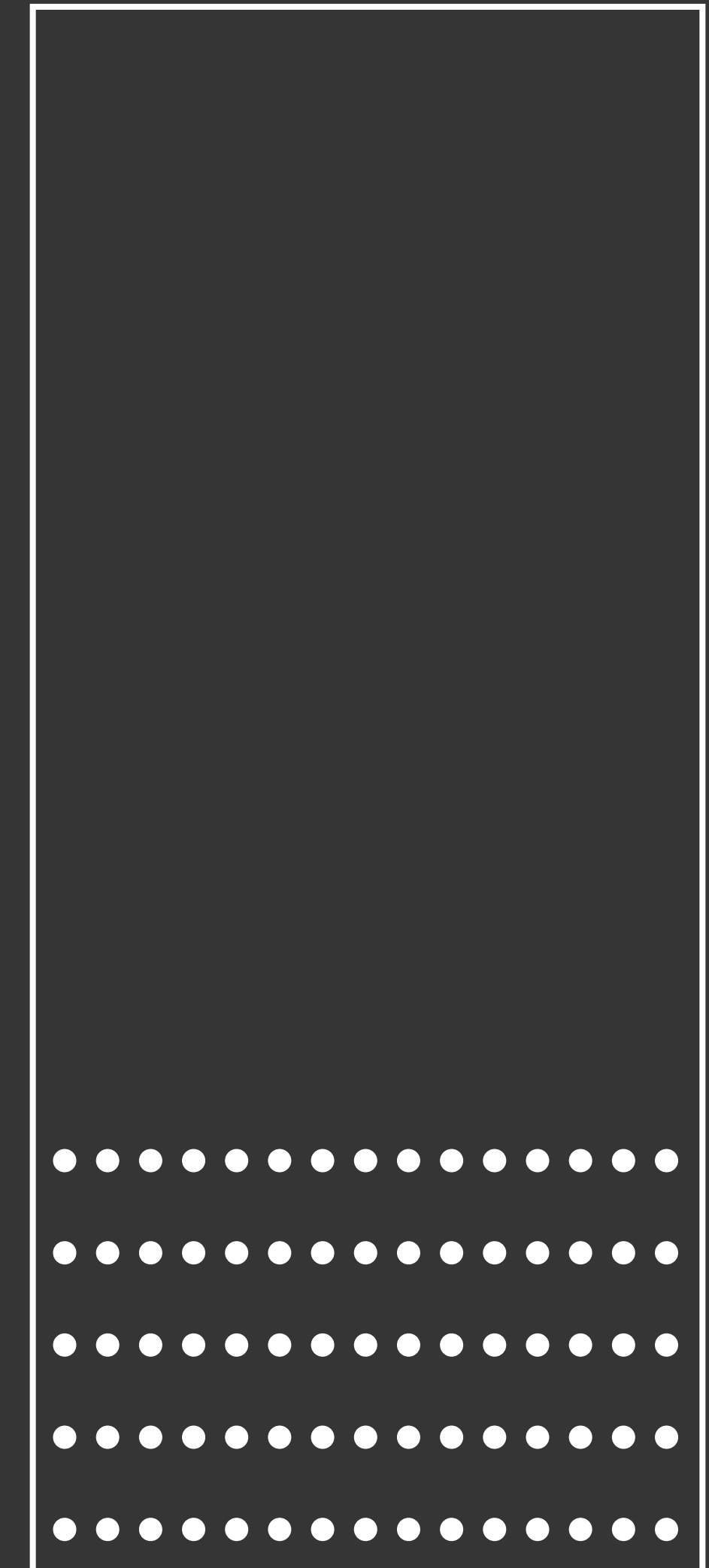
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();  
.....  
source.readUtf8LineStrict();  
.....  
source.readUtf8(40);  
.....  
.....  
source.readUtf8(40);  
.....  
.....  
source.readInt(); .....
```



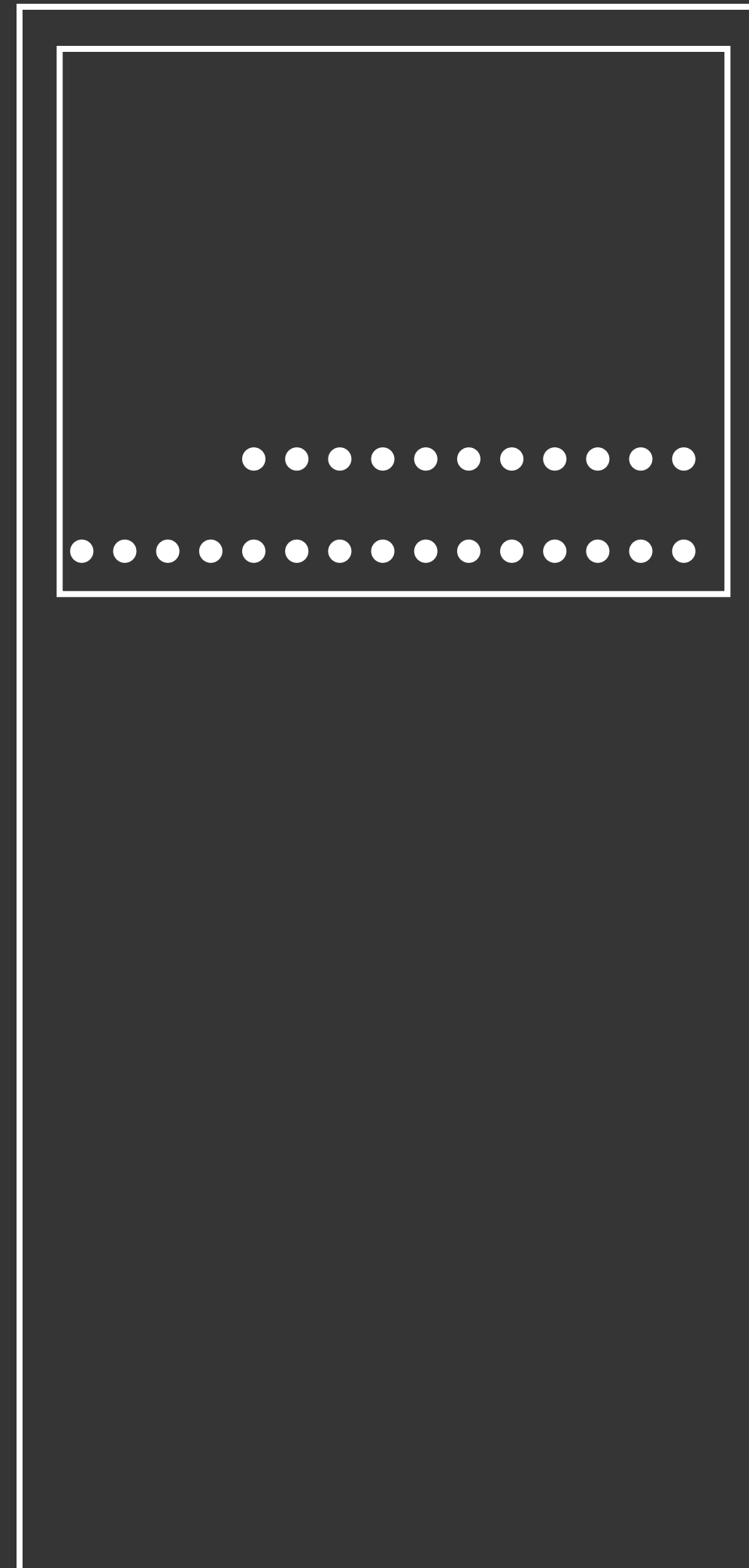
BufferedSource



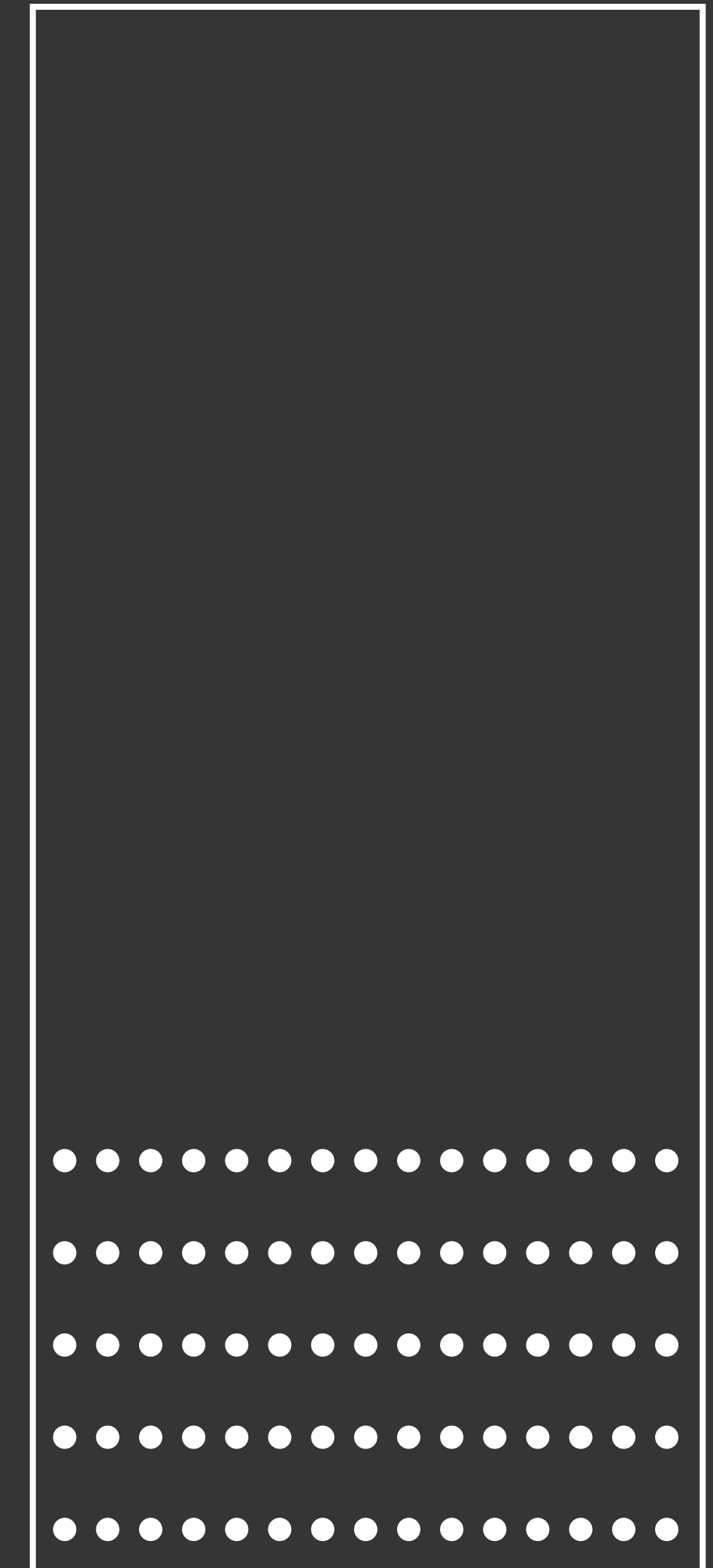
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();  
.....  
source.readUtf8LineStrict();  
.....  
source.readUtf8(40);  
.....  
.....  
source.readUtf8(40);  
.....  
.....  
source.readInt(); .....  
source.readInt();
```



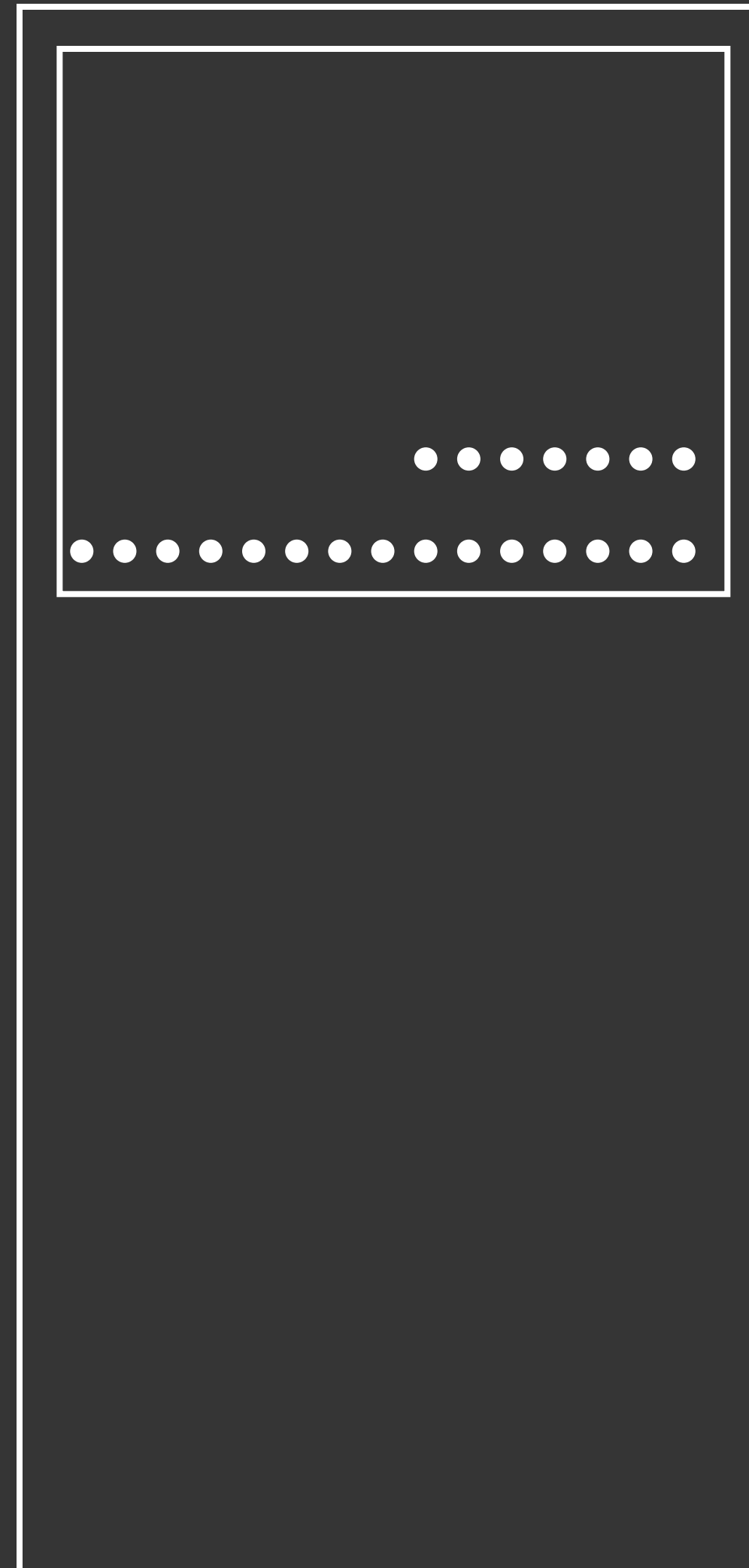
BufferedSource



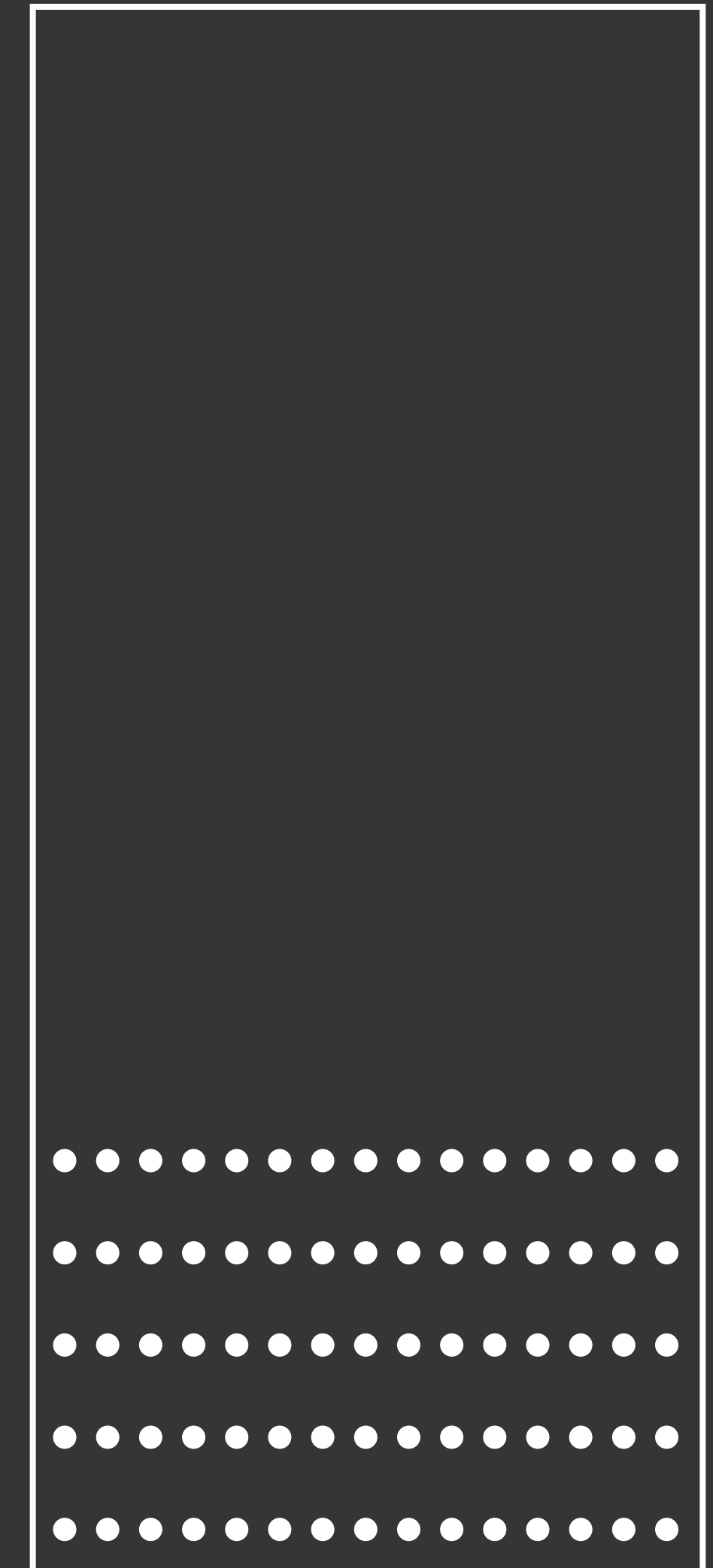
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();  
.....  
source.readUtf8LineStrict();  
.....  
source.readUtf8(40);  
.....  
.....  
source.readUtf8(40);  
.....  
.....  
source.readInt(); .....  
source.readInt(); .....
```



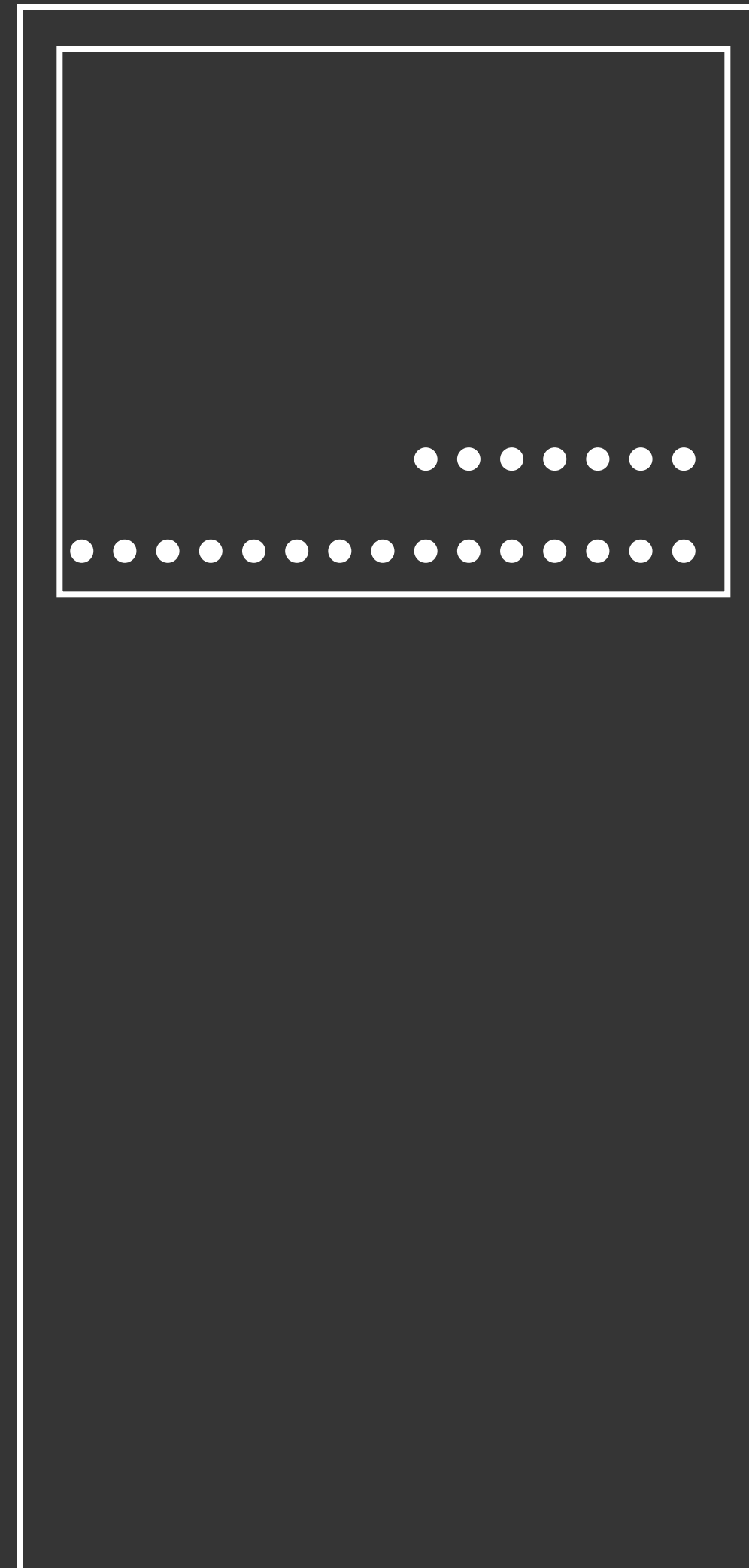
BufferedSource



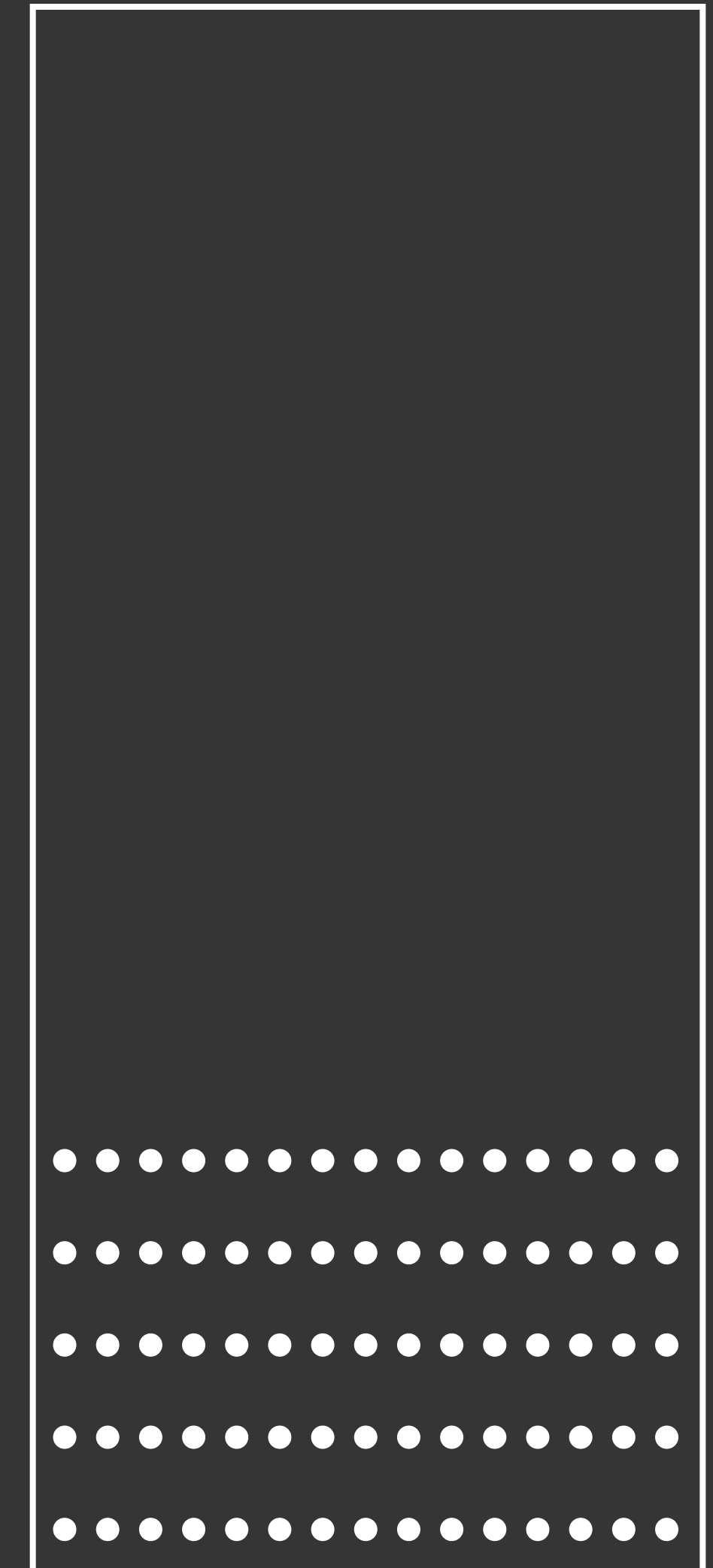
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();  
.....  
source.readUtf8LineStrict();  
.....  
source.readUtf8(40);  
.....  
.....  
source.readUtf8(40);  
.....  
.....  
source.readInt(); .....  
source.readInt(); .....  
source.readLong();
```



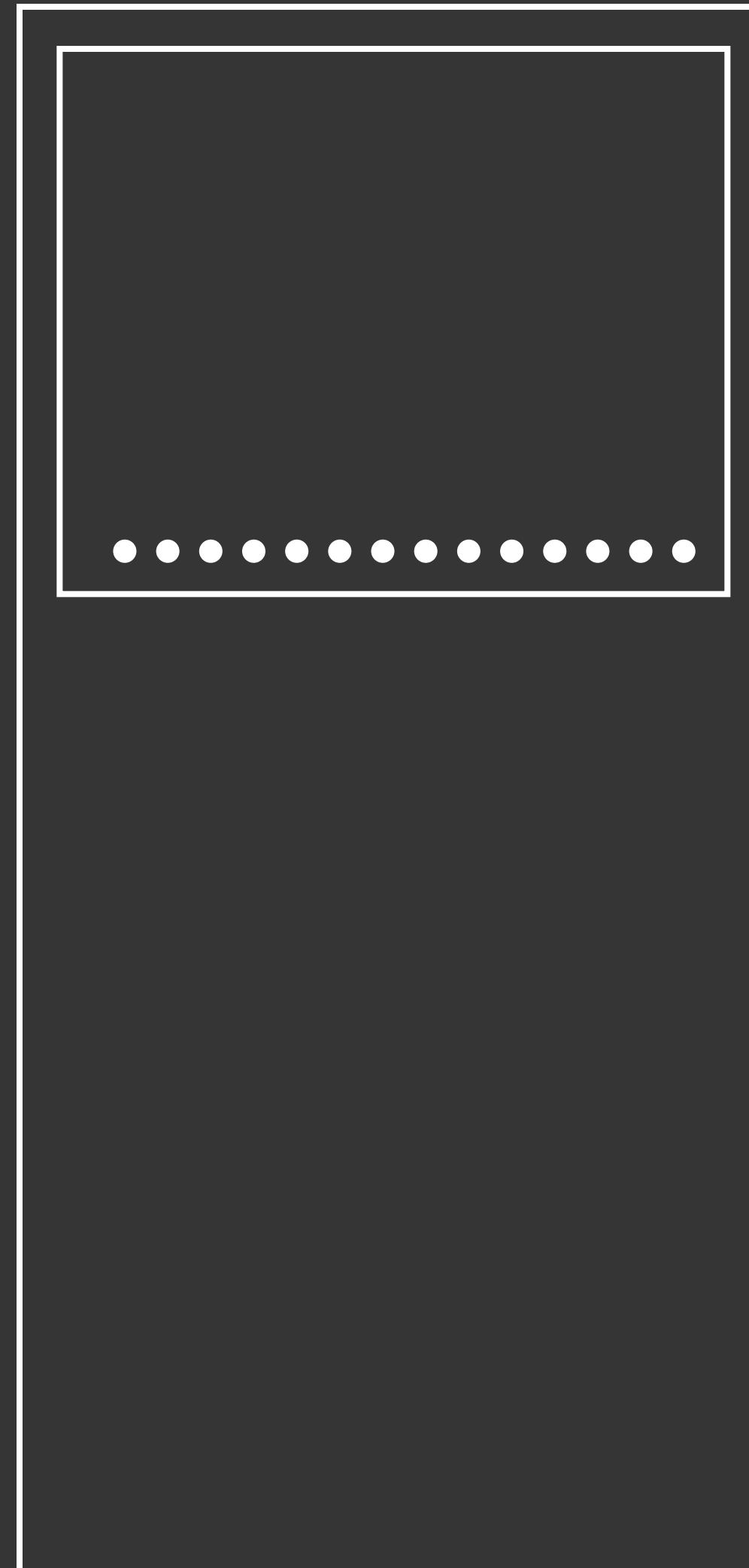
BufferedSource



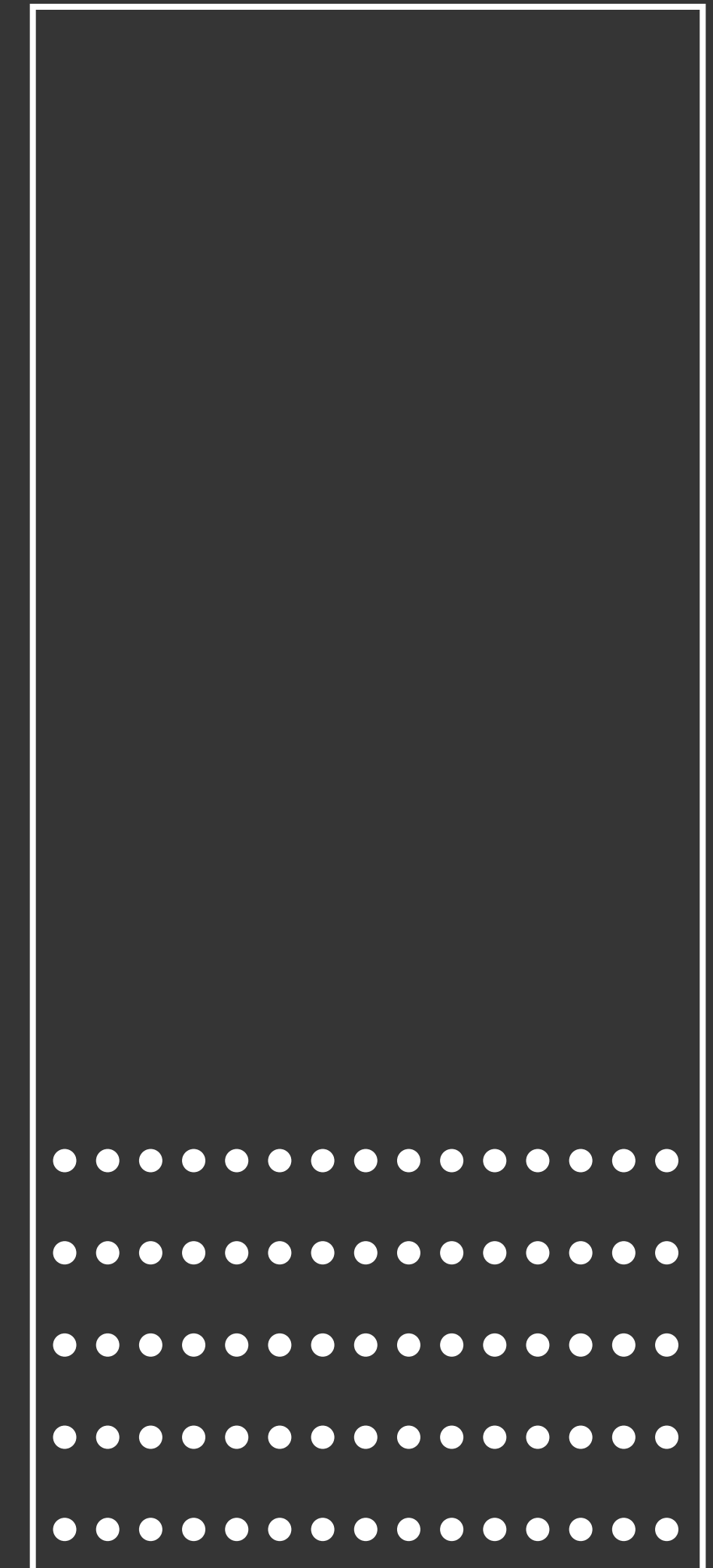
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();  
.....  
source.readUtf8LineStrict();  
.....  
source.readUtf8(40);  
.....  
.....  
source.readUtf8(40);  
.....  
.....  
source.readInt(); .....  
source.readInt(); .....  
source.readLong(); .....
```



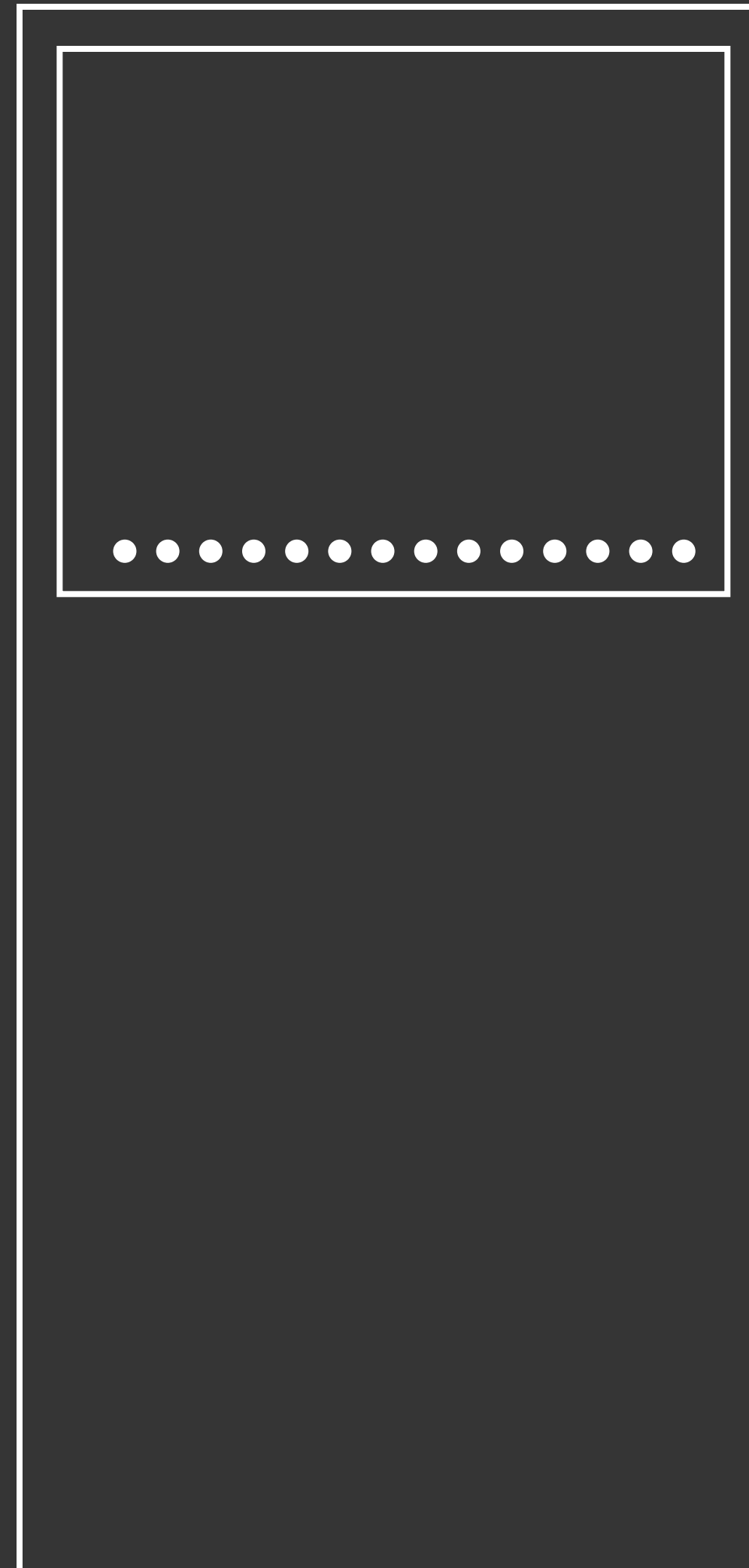
BufferedSource



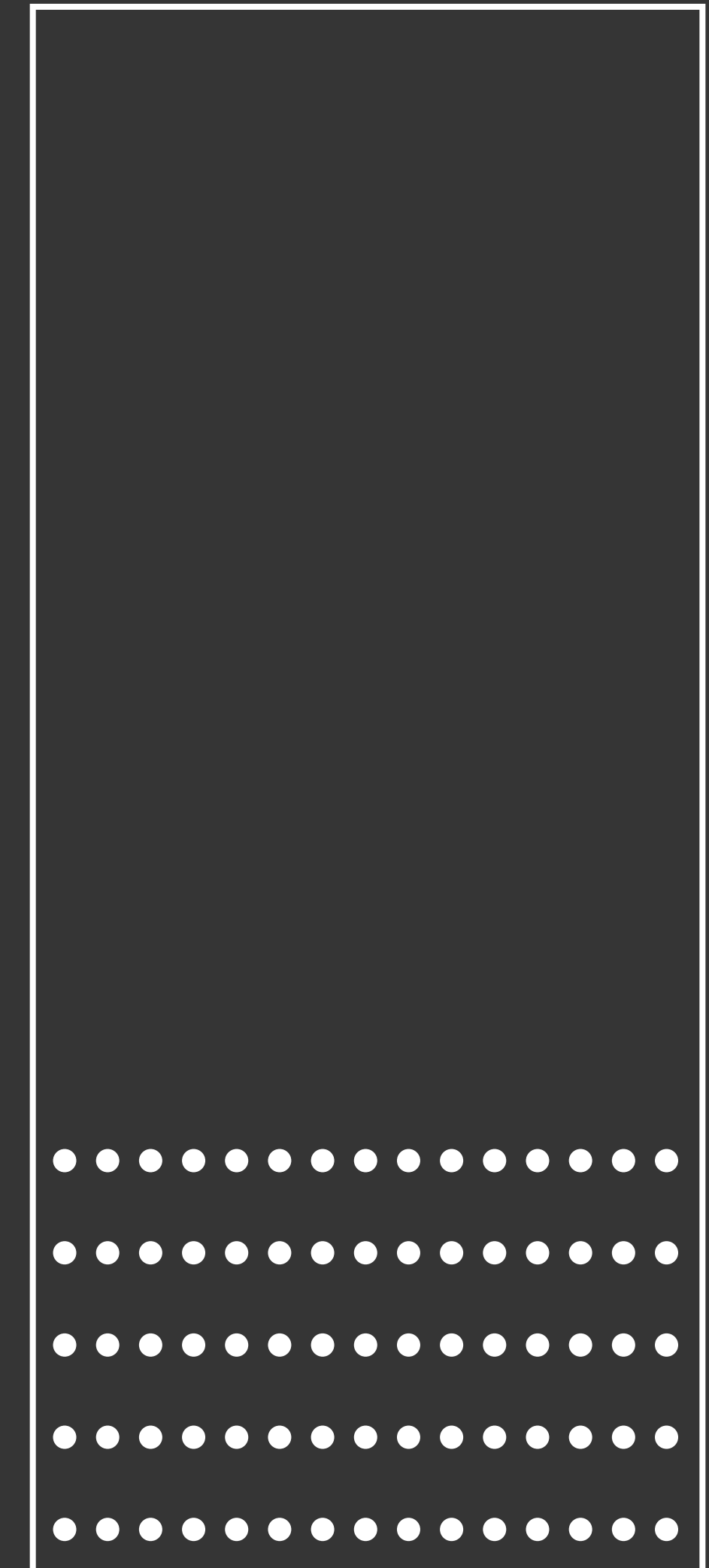
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();  
.....  
source.readUtf8LineStrict();  
.....  
source.readUtf8(40);  
.....  
.....  
source.readUtf8(40);  
.....  
.....  
source.readInt(); .....  
source.readInt(); .....  
source.readLong(); .....  
source.readLong();
```



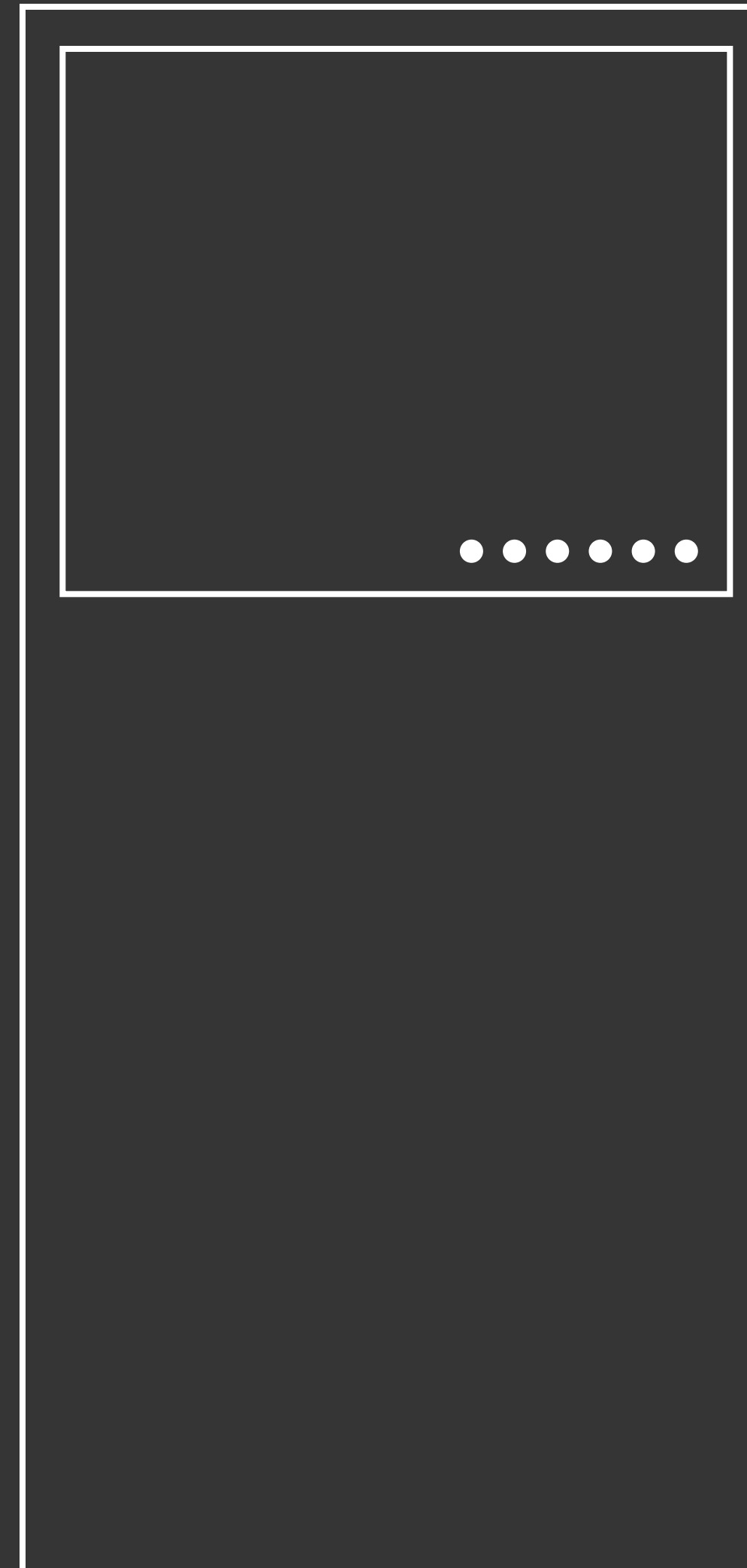
BufferedSource



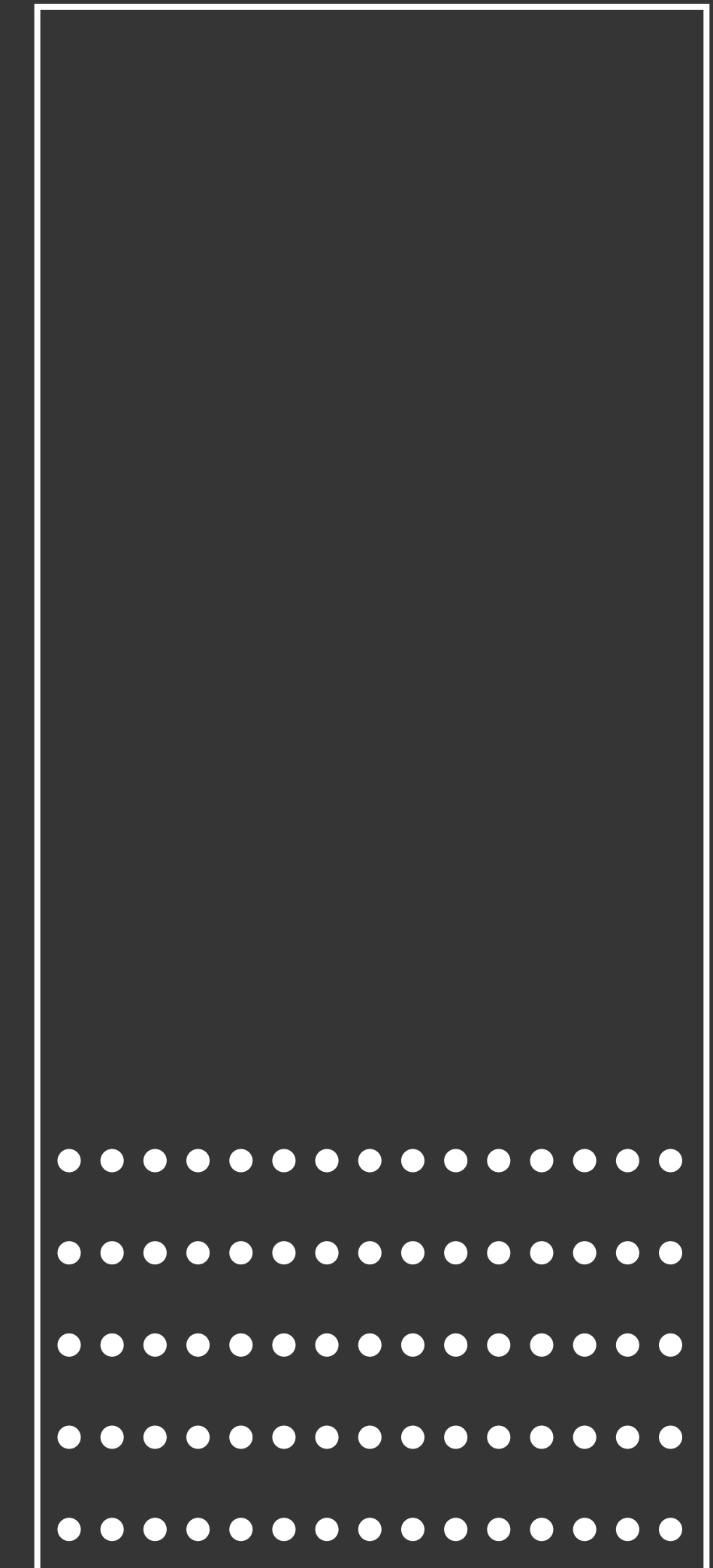
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();  
.....  
source.readUtf8LineStrict();  
.....  
source.readUtf8(40);  
.....  
.....  
source.readUtf8(40);  
.....  
.....  
source.readInt(); .....  
source.readInt(); .....  
source.readLong(); .....  
source.readLong(); .....
```



BufferedSource



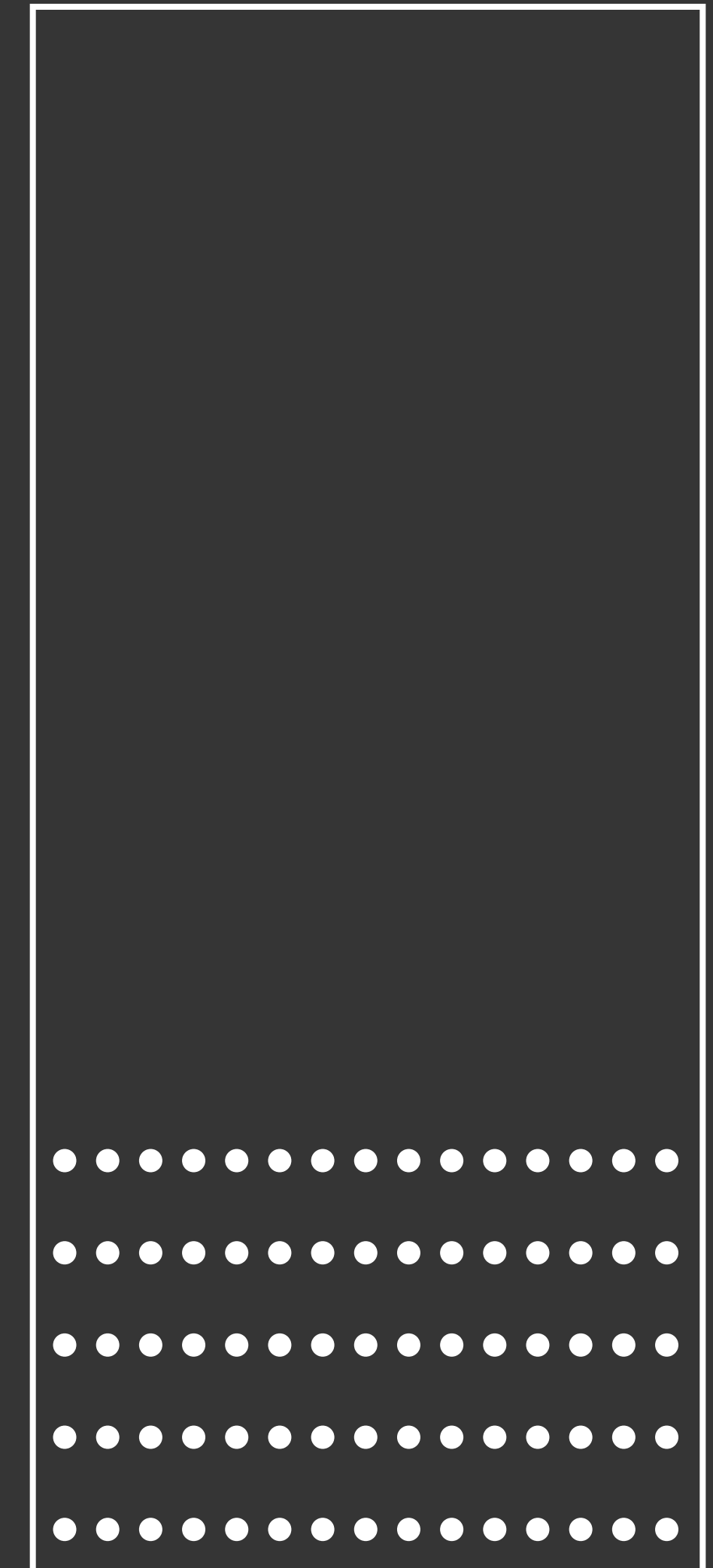
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();  
.....  
source.readUtf8LineStrict();  
.....  
source.readUtf8(40);  
.....  
.....  
source.readUtf8(40);  
.....  
.....  
source.readInt(); .....  
source.readInt(); .....  
source.readLong(); .....  
source.readLong(); .....  
source.readUtf8(6);
```



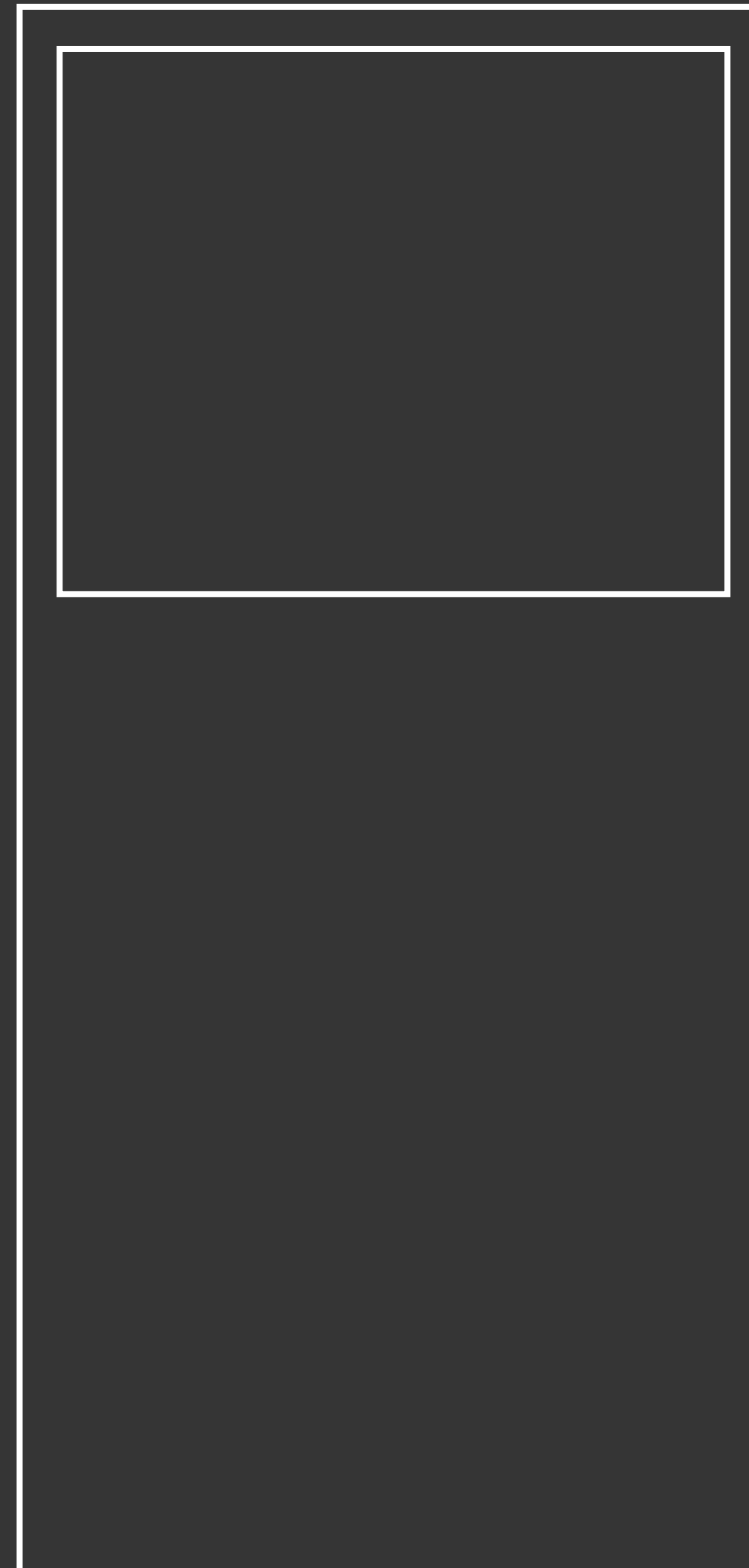
BufferedSource



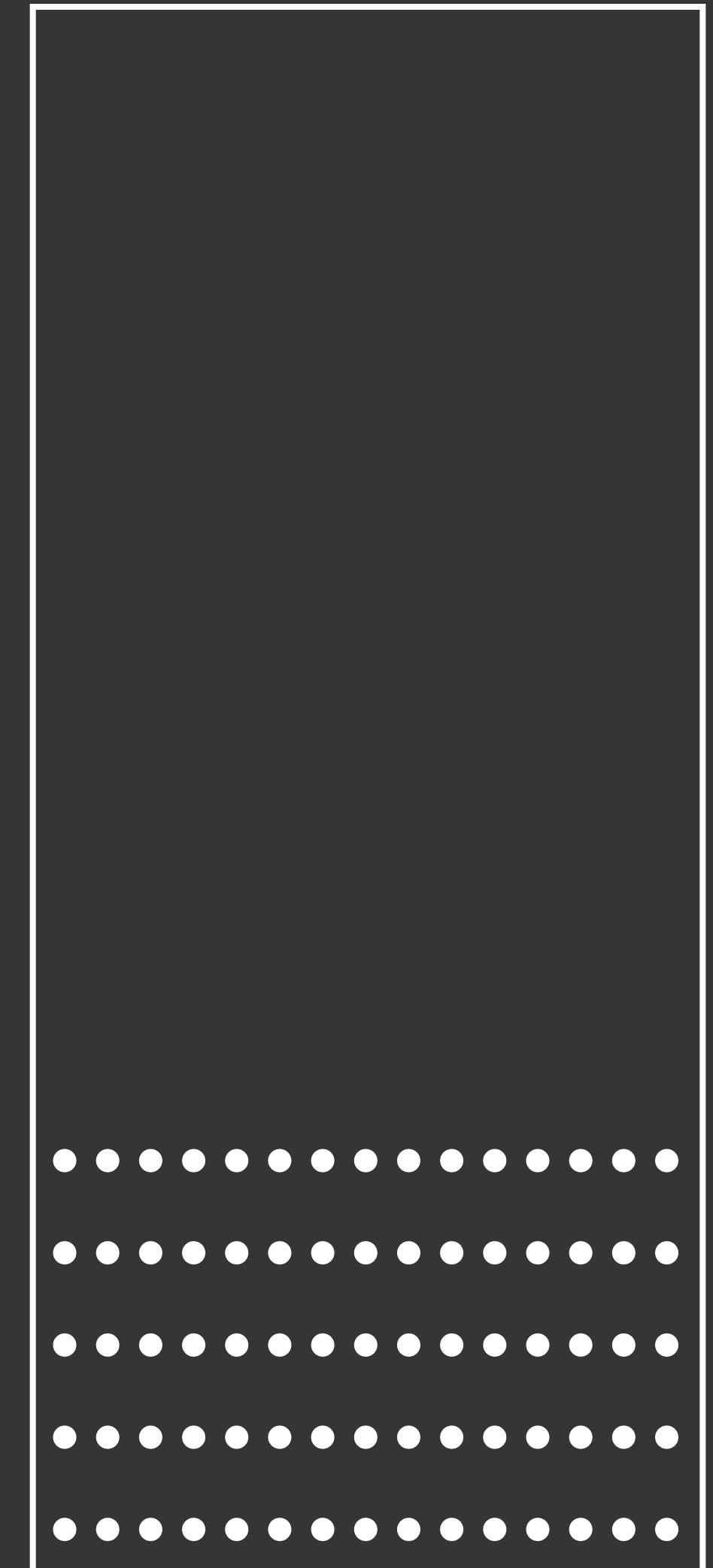
Socket

Okio

```
source.readInt(); .....
source.readInt(); .....
source.readUtf8(4); .....
source.readLong(); .....
source.readUtf8LineStrict();
.....
source.readUtf8LineStrict();
.....
source.readUtf8(40);
.....
.....
source.readUtf8(40);
.....
.....
source.readInt(); .....
source.readInt(); .....
source.readLong(); .....
source.readLong(); .....
source.readUtf8(6); .....
```



BufferedSource



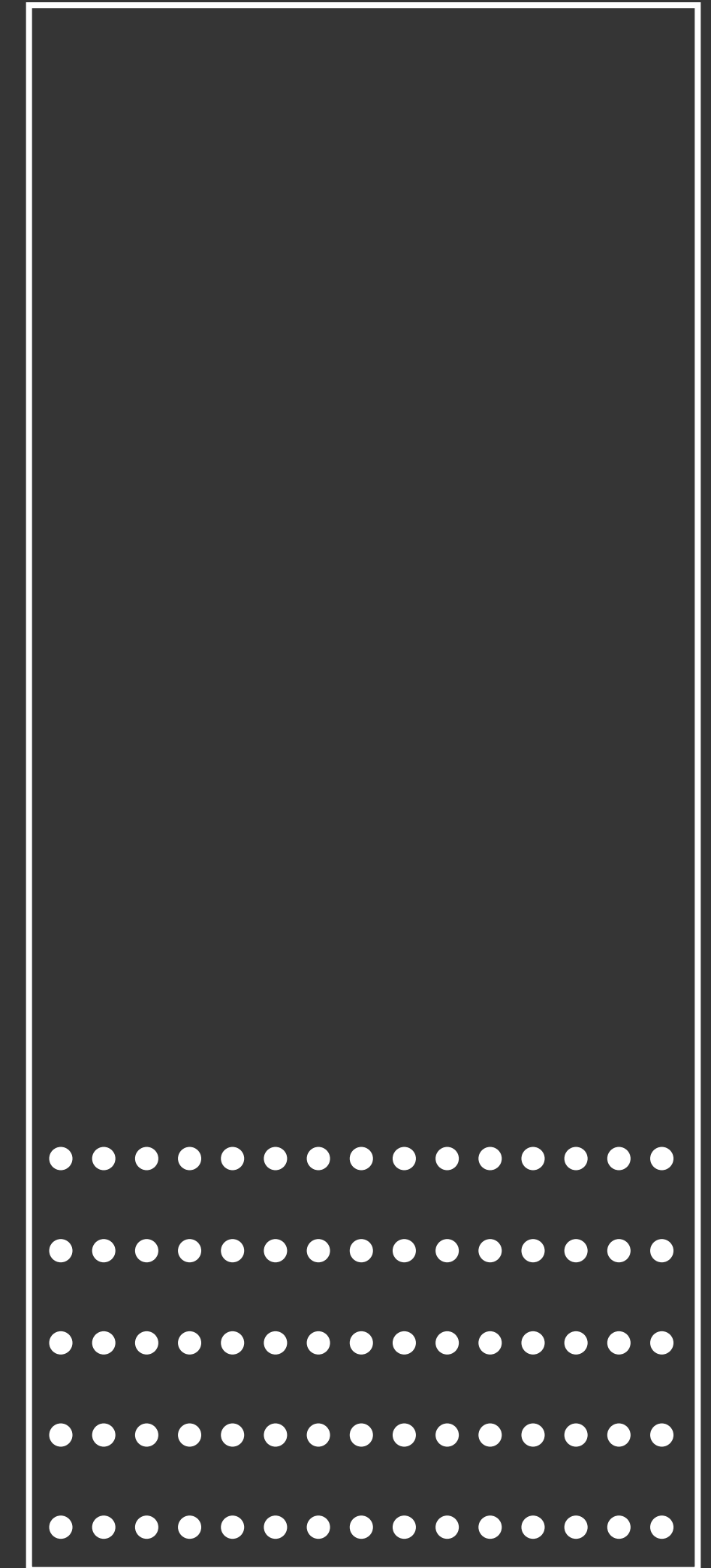
Socket

Okio

```
source.readInt(); .....  
source.readInt(); .....  
source.readUtf8(4); .....  
source.readLong(); .....  
source.readUtf8LineStrict();  
.....  
source.readUtf8LineStrict();  
.....  
source.readUtf8(40);  
.....  
.....  
source.readUtf8(40);  
.....  
.....  
source.readInt(); .....  
source.readInt(); .....  
source.readLong(); .....  
source.readLong(); .....  
source.readUtf8(6); .....
```



BufferedSource



Socket

Okio

```
class Okio {  
    static BufferedSource buffer(Source source) {}  
    static BufferedSink buffer(Sink sink) {}  
}
```

Okio

```
class Okio {  
    static BufferedSource buffer(Source source) {}  
    static BufferedSink buffer(Sink sink) {}  
  
    static Source source(InputStream in) {}  
    static Sink sink(OutputStream out) {}  
}
```

Okio

```
class Okio {  
    static BufferedSource buffer(Source source) {}  
    static BufferedSink buffer(Sink sink) {}  
  
    static Source source(InputStream in) {}  
    static Sink sink(OutputStream out) {}  
  
    static Source source(Socket socket) {}  
    static Sink sink(Socket socket) {}  
}
```

Okio

```
class Okio {  
    static BufferedSource buffer(Source source) {}  
    static BufferedSink buffer(Sink sink) {}  
  
    static Source source(InputStream in) {}  
    static Sink sink(OutputStream out) {}  
  
    static Source source(Socket socket) {}  
    static Sink sink(Socket socket) {}  
  
    static Source source(File file) {}  
    static Sink sink(File file) {}  
    static Sink appendingSink(File file) {}  
}
```


Okio

```
class Okio {  
    static BufferedSource buffer(Source source) {}  
    static BufferedSink buffer(Sink sink) {}  
  
    static Source source(InputStream in) {}  
    static Sink sink(OutputStream out) {}  
  
    static Source source(Socket socket) {}  
    static Sink sink(Socket socket) {}  
  
    static Source source(File file) {}  
    static Sink sink(File file) {}  
    static Sink appendingSink(File file) {}  
  
    static Source source(Path path, OpenOption... options) {}  
    static Sink sink(Path path, OpenOption... options) {}  
}
```

Okio

```
class DeflaterSink {  
    DeflaterSink(Sink sink, Deflater deflater) {}  
}
```

Okio

```
class DeflaterSink {  
    DeflaterSink(Sink sink, Deflater deflater) {}  
}
```

```
class InflaterSource {  
    InflaterSource(Source source, Inflater inflater) {}  
}
```

Okio

```
class DeflaterSink {  
    DeflaterSink(Sink sink, Deflater deflater) {}  
}
```

```
class InflaterSource {  
    InflaterSource(Source source, Inflater inflater) {}  
}
```

```
class GzipSink {  
    GzipSink(Sink sink) {}  
}
```

Okio

```
class DeflaterSink {  
    DeflaterSink(Sink sink, Deflater deflater) {}  
}
```

```
class InflaterSource {  
    InflaterSource(Source source, Inflater inflater) {}  
}
```

```
class GzipSink {  
    GzipSink(Sink sink) {}  
}
```

```
class GzipSource {  
    GzipSource(Source source) {}  
}
```

Okio

```
File file = // ...
```

File

Okio

```
File file = // ...  
Sink fileSink = Okio.sink(file);
```



Okio

```
File file = // ...  
Sink fileSink = Okio.sink(file);  
Sink gzipSink = new GzipSink(fileSink);
```



Okio

```
File file = // ...  
Sink fileSink = Okio.sink(file);  
Sink gzipSink = new GzipSink(fileSink);  
BufferedSink bufferedSink = Okio.buffer(gzipSink);
```



Okio

```
File file = // ...  
Sink fileSink = Okio.sink(file);  
Sink gzipSink = new GzipSink(fileSink);  
BufferedSink bufferedSink = Okio.buffer(gzipSink);  
  
bufferedSink.writeUtf8("Hello, hello, hello!");
```



Okio

```
File file = // ...  
Sink fileSink = Okio.sink(file);  
Sink gzipSink = new GzipSink(fileSink);  
BufferedSink bufferedSink = Okio.buffer(gzipSink);  
  
bufferedSink.writeUtf8("Hello, hello, hello!");
```

Hello, hello, hello!



Okio

```
File file = // ...  
Sink fileSink = Okio.sink(file);  
Sink gzipSink = new GzipSink(fileSink);  
BufferedSink bufferedSink = Okio.buffer(gzipSink);  
  
bufferedSink.writeUtf8("Hello, hello, hello!");  
bufferedSink.close();
```

Hello, hello, hello!



Okio

```
File file = // ...  
Sink fileSink = Okio.sink(file);  
Sink gzipSink = new GzipSink(fileSink);  
BufferedSink bufferedSink = Okio.buffer(gzipSink);  
  
bufferedSink.writeUtf8("Hello, hello, hello!");  
bufferedSink.close();
```



Okio

```
File file = // ...  
Sink fileSink = Okio.sink(file);  
Sink gzipSink = new GzipSink(fileSink);  
BufferedSink bufferedSink = Okio.buffer(gzipSink);  
  
bufferedSink.writeUtf8("Hello, hello, hello!");  
bufferedSink.close();
```



Okio

```
File file = // ...  
Sink fileSink = Okio.sink(file);  
Sink gzipSink = new GzipSink(fileSink);  
BufferedSink bufferedSink = Okio.buffer(gzipSink);  
  
bufferedSink.writeUtf8("Hello, hello, hello!");  
bufferedSink.close();
```



Okio

```
File file = // ...  
Sink fileSink = Okio.sink(file);  
Sink gzipSink = new GzipSink(fileSink);  
Sha1Sink hashingSink = new Sha1Sink(gzipSink);  
BufferedSink bufferedSink = Okio.buffer(hashingSink);  
  
bufferedSink.writeUtf8("Hello, hello, hello!");  
bufferedSink.close();  
  
String hash = hashingSink.hash();
```



Okio

```
Socket socket = //...  
Source socketSource = Okio.source(socket);
```

Okio

```
Socket socket = //...  
Source socketSource = Okio.source(socket);  
  
Throttler throttler = new Throttler();  
Source throttledSource = throttler.throttle(socketSource);
```

Okio

```
Socket socket = //...  
Source socketSource = Okio.source(socket);  
  
Throttler throttler = new Throttler();  
Source throttledSource = throttler.throttle(socketSource);  
  
BufferedSource bufferedSource = Okio.buffer(throttledSource);  
bufferedSource.readUtf8(10 * 1024);
```

Okio

```
Socket socket = //...
Source socketSource = Okio.source(socket);

Throttler throttler = new Throttler();
Source throttledSource = throttler.throttle(socketSource);

BufferedSource bufferedSource = Okio.buffer(throttledSource);
bufferedSource.readUtf8(10 * 1024);

throttler.bytesPerTimePeriod(50, 1, SECONDS);
bufferedSource.readUtf8(10 * 1024);
```

OkHttp

OkHttp

```
OkHttpClient client = new OkHttpClient();
```

OkHttp

```
OkHttpClient client = new OkHttpClient();  
  
Request request = new Request.Builder()  
    .url("http://droidcon.ca/a-few-ok-libraries.html")  
    .build();
```

OkHttp

```
OkHttpClient client = new OkHttpClient();

RequestBody requestBody = // ...
Request request = new Request.Builder()
    .url("http://droidcon.ca/a-few-ok-libraries.html")
    .post(requestBody)
    .build();
```


OkHttp

```
OkHttpClient client = new OkHttpClient();
```

```
RequestBody requestBody = // ...  
Request request = new Request.Builder()  
    .url("http://droidcon.ca/a-few-ok-libraries.html")  
    .post(requestBody)  
    .build();
```

```
class RequestBody {  
    static RequestBody create(MediaType mediaType, String body) {}  
    static RequestBody create(MediaType mediaType, byte[] body) {}  
    static RequestBody create(MediaType mediaType, File body) {}  
}
```

OkHttp

```
class RequestBody {  
    static RequestBody create(MediaType mediaType, String body) {}  
    static RequestBody create(MediaType mediaType, byte[] body) {}  
    static RequestBody create(MediaType mediaType, File body) {}  
}
```

OkHttp

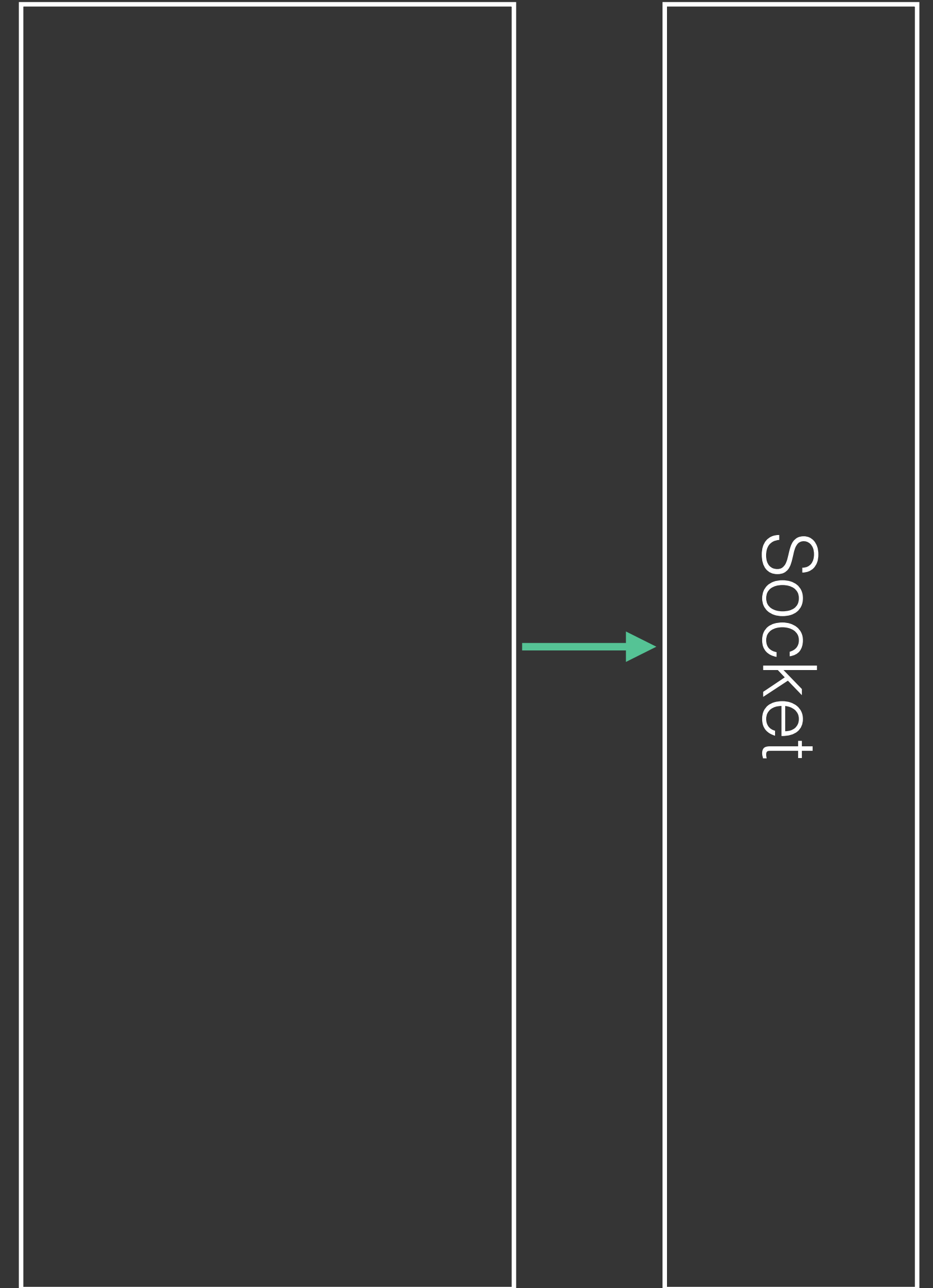
```
abstract class RequestBody {  
    static RequestBody create(MediaType mediaType, String body) {}  
    static RequestBody create(MediaType mediaType, byte[] body) {}  
    static RequestBody create(MediaType mediaType, File body) {}  
  
    long contentLength();  
    MediaType contentType();  
    void writeTo(BufferedSink sink);  
}
```

OkHttp

```
requestBody.writeTo(sink);
```

OkHttp

```
requestBody.writeTo(sink);
```



OkHttp

```
requestBody.writeTo(sink);
```

```
GET /a-few-ok-libraries.html HTTP/1.1  
Host: droidcon.ca  
Content-Type: text/plain  
Content-Length: 3000
```

Socket

OkHttp

```
requestBody.writeTo(sink);
```

FixedLengthSink

```
GET /a-few-ok-libraries.html HTTP/1.1  
Host: droidcon.ca  
Content-Type: text/plain  
Content-Length: 3000
```

Socket

OkHttp

```
requestBody.writeTo(sink);
```

FixedLengthSink

```
GET /a-few-ok-libraries.html HTTP/1.1
Host: droidcon.ca
Content-Type: text/plain
Content-Length: 3000
```

You think water moves fast? You should see ice. It moves like it has a mind. Like it knows it killed the world once and got a taste for murder. After the avalanche, it took us a week to climb out. Now, I don't know exactly when we turned on each other, but I know that seven of us survived the slide... and only five made it out. Now we took an oath, that I'm breaking now. We said we'd say it was the snow that killed the other two, but it wasn't. Nature is lethal but it doesn't hold a candle to man.

Your bones don't break, mine do. That's clear. Your cells react to bacteria and viruses differently than mine. You don't get sick, I do. That's also clear. But for some reason, you and I react the exact same way to water. We swallow it too fast, we choke. We get some in our lungs, we drown. However unreal it may seem, we are connected, you and I. We're on the same curve, just on opposite ends.

Normally, both your asses would be dead as fucking fried chicken, but you happen to pull this shit while I'm in a transitional period so I don't wanna kill you, I wanna help you. But I can't give you this case, it don't belong to me. Besides, I've already been through too much shit this morning over this case to hand it over to your dumb ass.

The path of the righteous man is beset on all sides by the iniquities of the selfish and the tyranny of evil men. Blessed is he who, in the name of charity and good will, shepherds the weak through the valley of darkness, for he is truly his brother's keeper and the finder of lost children. And I will strike down upon thee with great vengeance and furious anger those who would attempt to poison and destroy My brothers. And you will know My name is the Lord when I lay My vengeance upon thee.

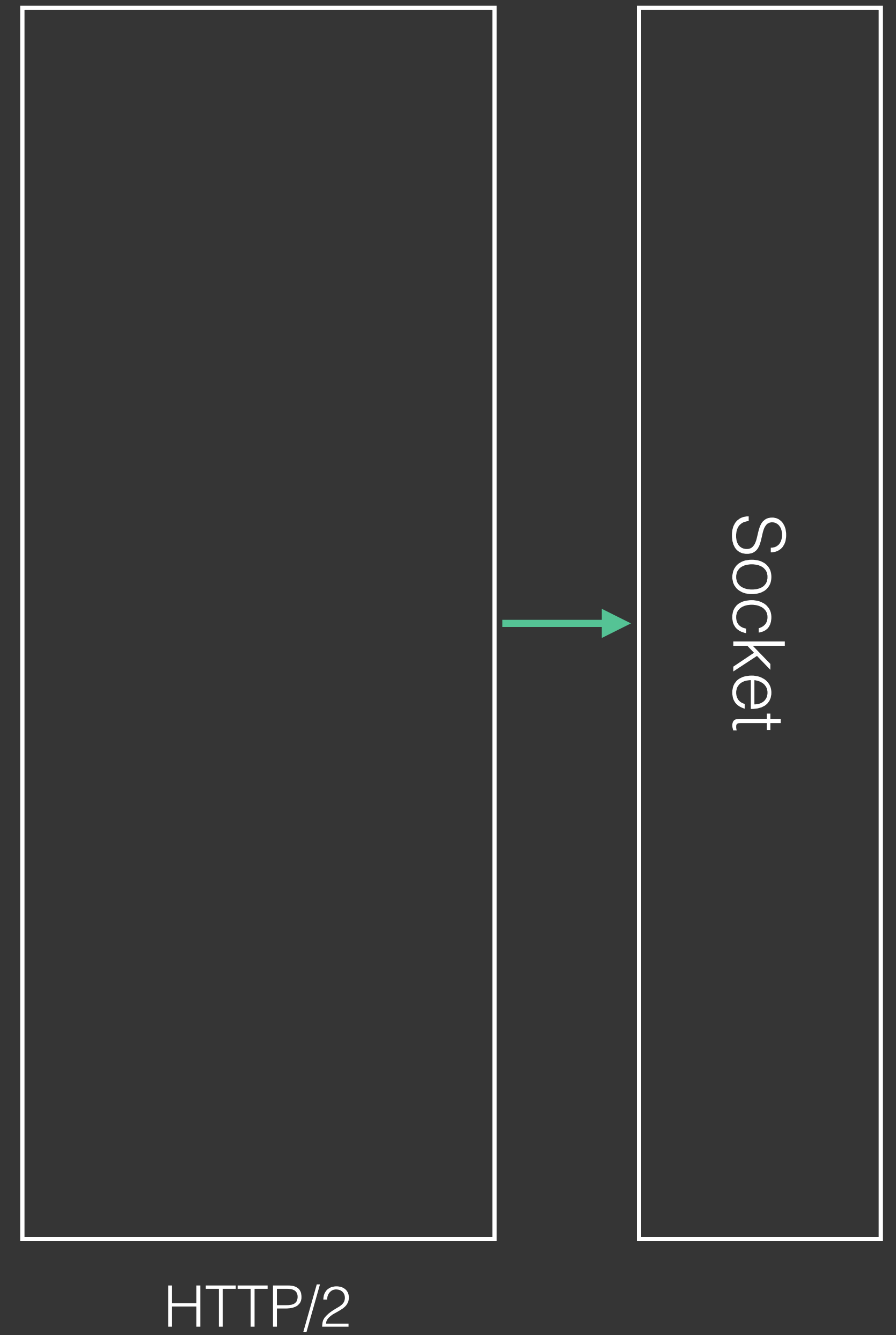
My money's in that office, right? If she start giving me some bullshit about it ain't there, and we got to go someplace else and get it, I'm gonna shoot you in the head then and there. Then I'm gonna shoot that bitch in the kneecaps, find out where my goddamn money is. She gonna tell me too. Hey, look at me when I'm talking to you, motherfucker. You listen: we go in there, and that nigga Winston or anybody else is in there, you the first motherfucker to get shot. You understand?

Look, just because I don't be givin' no man a foot massage don't make it right for Marsellus to throw Antwone into a glass motherfuckin' house, fuckin' up the way the nigger talks. Motherfucker do that shit to me, he better paralyze my ass, 'cause I'll kill the motherfucker, know what I'm sayin'?

Well, the way they make shows is, they make one show. That show's called a pilot. Then they show that show to the people who make shows, and on the strength of that one show they decide if they're going to make more shows. Some pilots get picked and become television programs. Some don't, become nothing. She starred in one of the ones that became nothing.

Socket

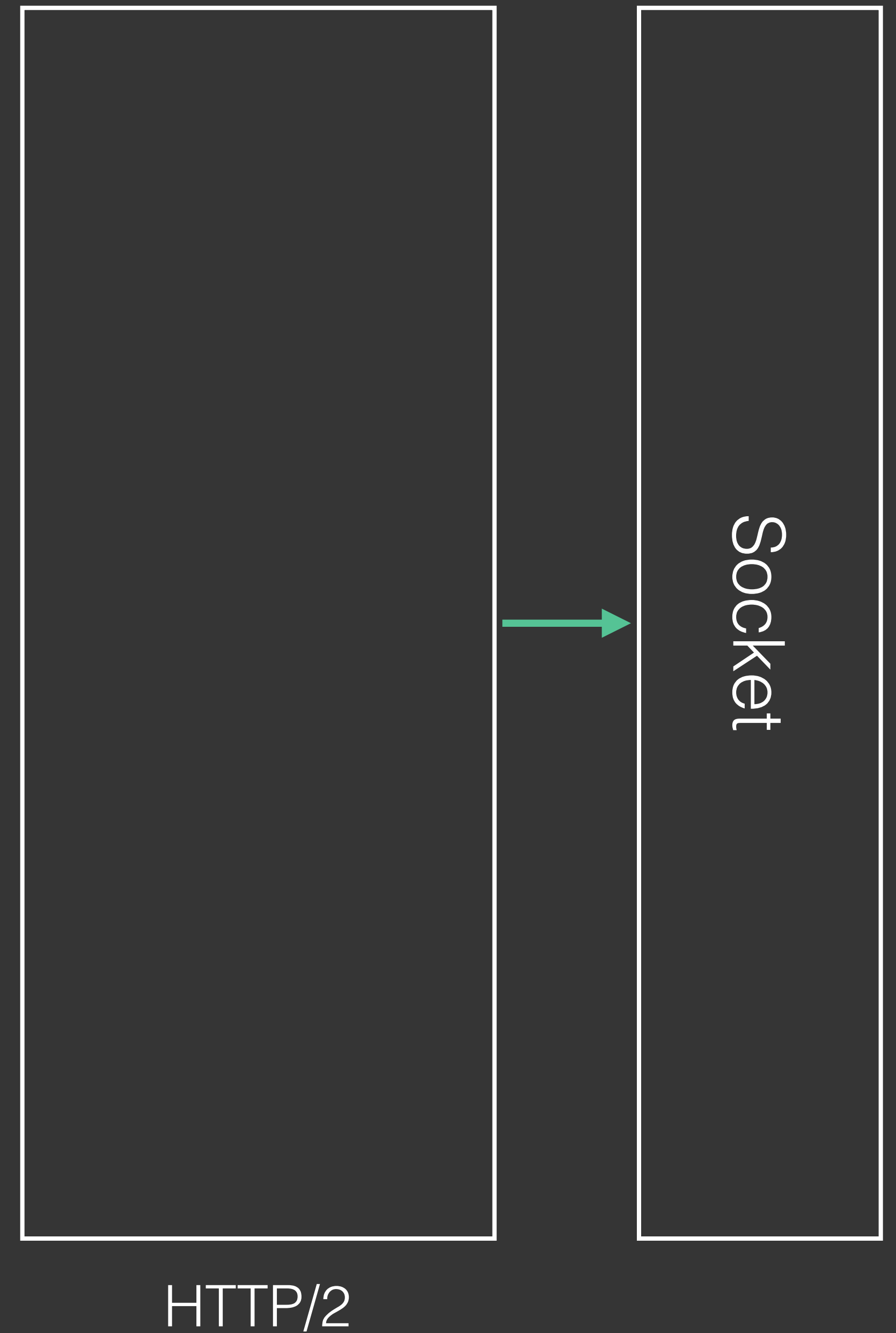
OkHttp



OkHttp

requestBody1

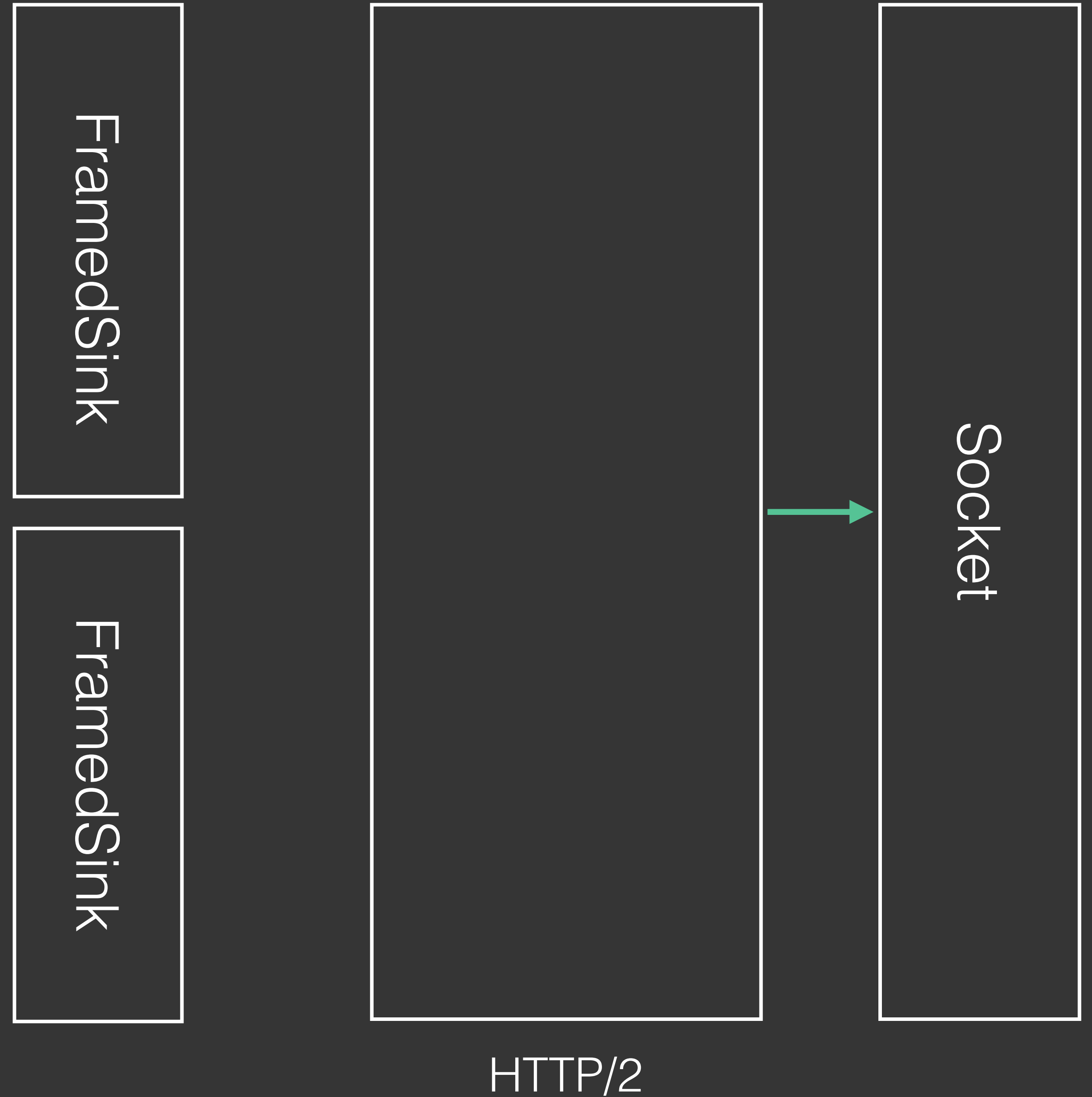
requestBody2



OkHttp

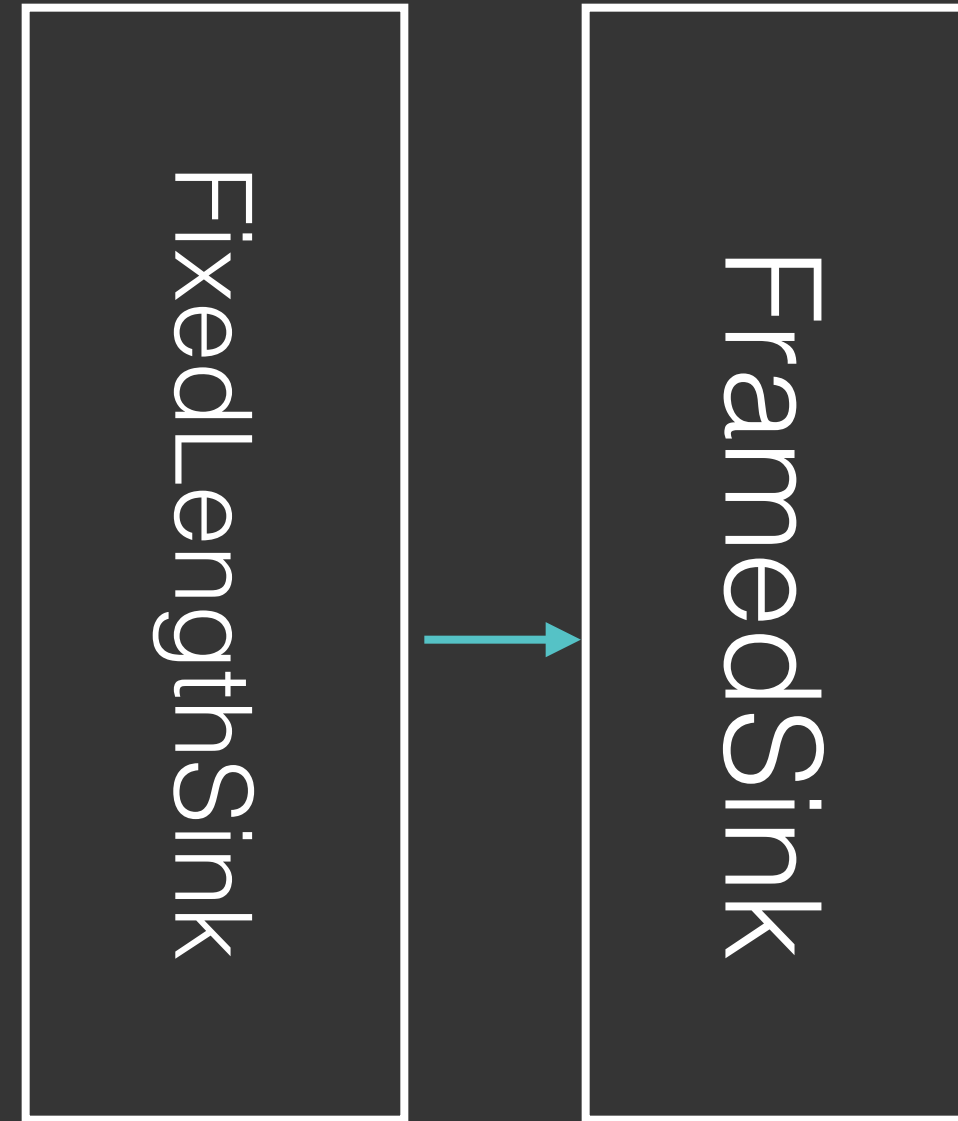
requestBody1

requestBody2

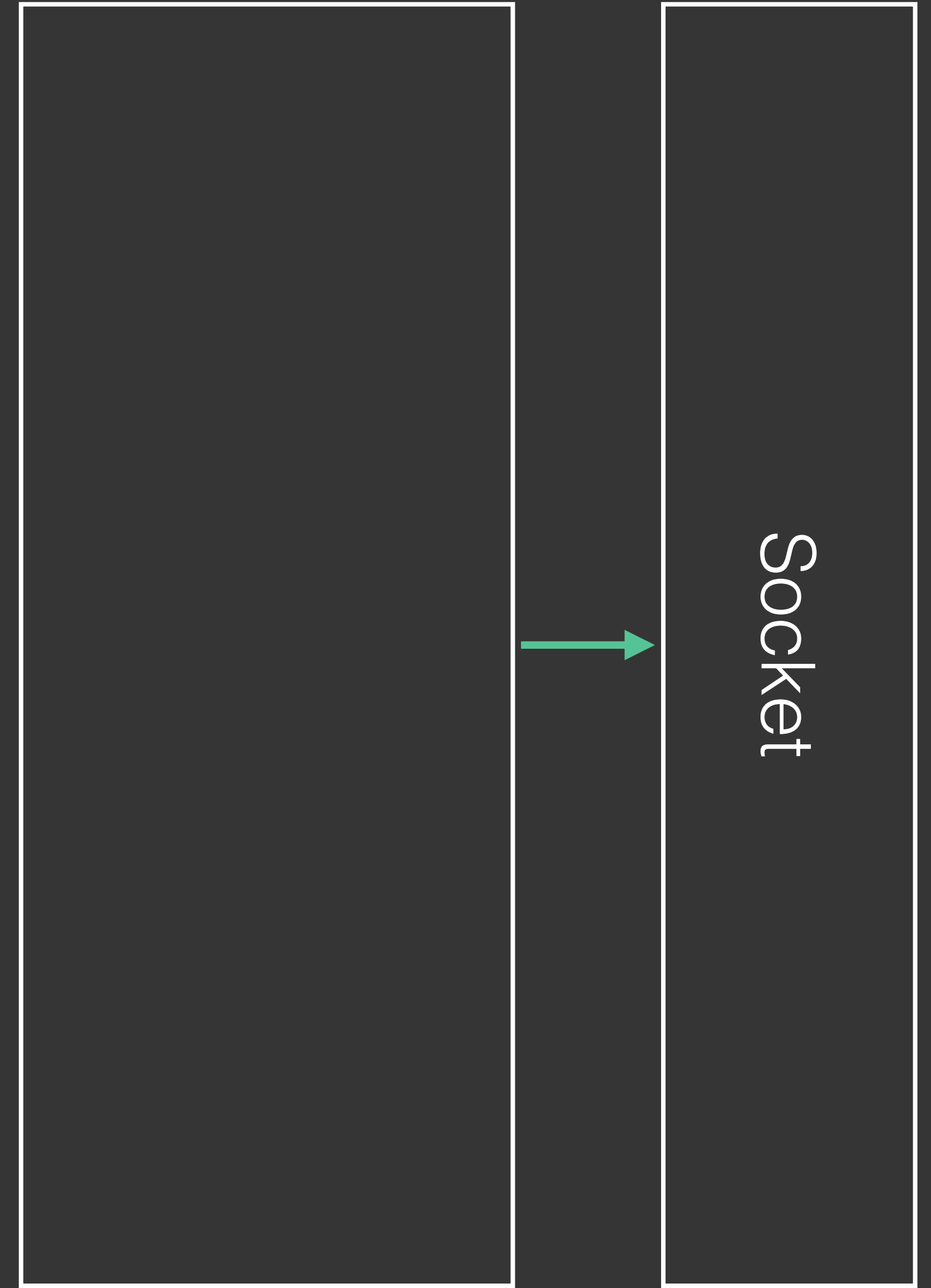
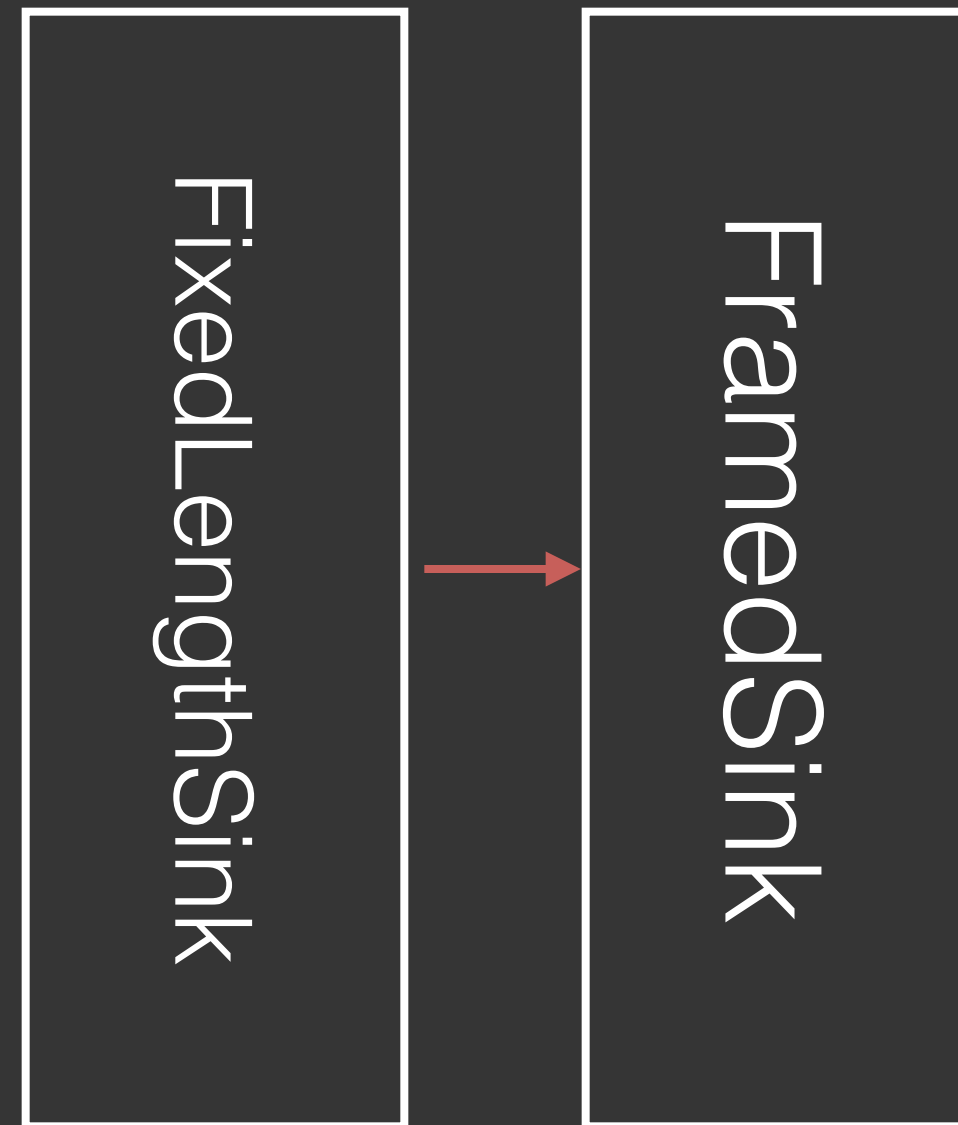


OkHttp

requestBody1



requestBody2



HTTP/2

OkHttp

requestBody1 →

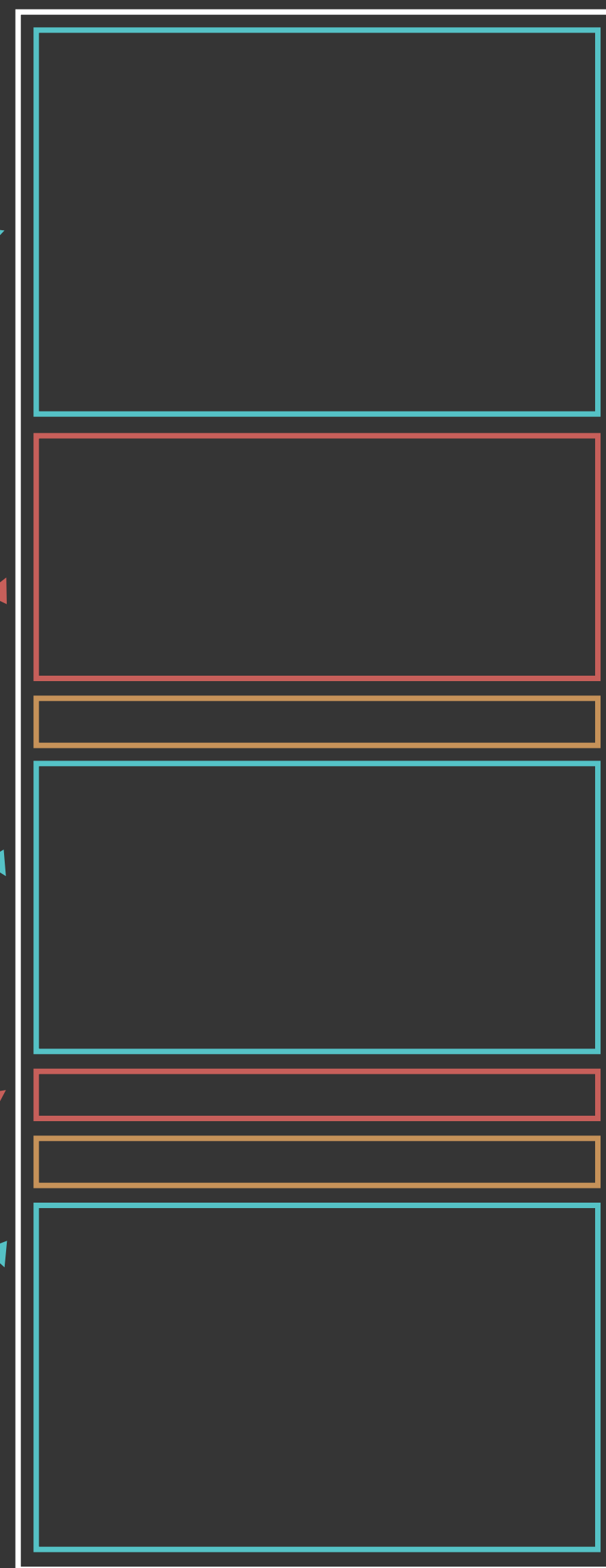
FixedLengthSink

FramedSink

requestBody2 →

FixedLengthSink

FramedSink



Socket

HTTP/2

OkHttp

```
OkHttpClient client = new OkHttpClient();  
  
Request request = new Request.Builder()  
    .url("http://droidcon.ca/a-few-ok-libraries.html")  
    .build();
```

OkHttp

```
OkHttpClient client = new OkHttpClient();

Request request = new Request.Builder()
    .url("http://droidcon.ca/a-few-ok-libraries.html")
    .build();

Response response = client.newCall(request).execute();
```

OkHttp

```
OkHttpClient client = new OkHttpClient();

Request request = new Request.Builder()
    .url("http://droidcon.ca/a-few-ok-libraries.html")
    .build();

Response response = client.newCall(request).execute();

System.out.println(response.body().string());
```


OkHttp

```
OkHttpClient client = new OkHttpClient();

Request request = new Request.Builder()
    .url("http://droidcon.ca/a-few-ok-libraries.html")
    .build();

client.newCall(request).enqueue(new Callback() {
    @Override void onResponse(Response response) {
        System.out.println(response.body().string());
    }

    @Override void onFailure(Request request, IOException e) {
        // ...
    }
});
```

OkHttp

```
String string = response.body().string();
```

OkHttp

```
ResponseBody body = response.body();
```

```
String string = body.string();
```

OkHttp

```
ResponseBody body = response.body();
```

```
String string = body.string();  
Reader reader = body.charStream();
```

OkHttp

```
ResponseBody body = response.body();  
  
String string = body.string();  
Reader reader = body.charStream();  
InputStream stream = body.byteStream();
```

OkHttp

```
ResponseBody body = response.body();  
  
String string = body.string();  
Reader reader = body.charStream();  
InputStream stream = body.byteStream();  
byte[] bytes = body.bytes();
```

OkHttp

```
ResponseBody body = response.body();
```

```
String string = body.string();
```

```
Reader reader = body.charStream();
```

```
InputStream stream = body.byteStream();
```

```
byte[] bytes = body.bytes();
```

```
BufferedSource source = body.source();
```

OkHttp

```
body.source();
```


OkHttp

```
body.source();
```

```
HTTP/1.1 200 OK
Date: Thu, 9 Apr 2015 15:14:34 GMT
Server: JakeWharton/2.3.7 (Human)
Content-Type: application/presentation; charset=UTF-8
Content-Length: 3000
```

You think water moves fast? You should see ice. It moves like it has a mind. Like it knows it killed the world once and got a taste for murder. After the avalanche, it took us a week to climb out. Now, I don't know exactly when we turned on each other, but I know that seven of us survived the slide... and only five made it out. Now we took an oath, that I'm breaking now. We said we'd say it was the snow that killed the other two, but it wasn't. Nature is lethal but it doesn't hold a candle to man.

Your bones don't break, mine do. That's clear. Your cells react to bacteria and viruses differently than mine. You don't get sick, I do. That's also clear. But for some reason, you and I react the exact same way to water. We swallow it too fast, we choke. We get some in our lungs, we drown. However unreal it may seem, we are connected, you and I. We're on the same curve, just on opposite ends.

Normally, both your asses would be dead as fucking fried chicken, but you happen to pull this shit while I'm in a transitional period so I don't wanna kill you, I wanna help you. But I can't give you this case, it don't belong to me. Besides, I've already been through too much shit this morning over this case to hand it over to your dumb ass.

The path of the righteous man is beset on all sides by the iniquities of the selfish and the tyranny of evil men. Blessed is he who, in the name of charity and good will, shepherds the weak through the valley of darkness, for he is truly his brother's keeper and the finder of lost children. And I will strike down upon thee with great vengeance and furious anger those who would attempt to poison and destroy My brothers. And you will know My name is the Lord when I lay My vengeance upon thee.

My money's in that office, right? If she start giving me some bullshit about it ain't there, and we got to go someplace else and get it, I'm gonna shoot you in the head then and there. Then I'm gonna shoot that bitch in the kneecaps, find out where my goddamn money is. She gonna tell me too. Hey, look at me when I'm talking to you, motherfucker. You listen: we go in there, and that nigga Winston or anybody else is in there, you the first motherfucker to get shot. You understand?

Look, just because I don't be givin' no man a foot massage don't make it right for Marsellus to throw Antwone into a glass motherfuckin' house, fuckin' up the way the nigger talks. Motherfucker do that shit to me, he better paralyze my ass, 'cause I'll kill the motherfucker, know what I'm sayin'?

Well, the way they make shows is, they make one show. That show's called a pilot. Then they show that show to the people who make shows, and on the strength of that one show they decide if they're going to make more shows. Some pilots get picked and become television programs. Some don't, become nothing. She starred in one of the ones that became nothing.

Socket

OkHttp

```
HTTP/1.1 200 OK
Date: Thu, 9 Apr 2015 15:14:34 GMT
Server: JakeWharton/2.3.7 (Human)
Content-Type: application/presentation; charset=UTF-8
Content-Length: 3000
```

```
body.source();
```

```
HTTP/1.1 200 OK
Date: Thu, 9 Apr 2015 15:14:34 GMT
Server: JakeWharton/2.3.7 (Human)
Content-Type: application/presentation; charset=UTF-8
Content-Length: 3000
```

You think water moves fast? You should see ice. It moves like it has a mind. Like it knows it killed the world once and got a taste for murder. After the avalanche, it took us a week to climb out. Now, I don't know exactly when we turned on each other, but I know that seven of us survived the slide... and only five made it out. Now we took an oath, that I'm breaking now. We said we'd say it was the snow that killed the other two, but it wasn't. Nature is lethal but it doesn't hold a candle to man.

Your bones don't break, mine do. That's clear. Your cells react to bacteria and viruses differently than mine. You don't get sick, I do. That's also clear. But for some reason, you and I react the exact same way to water. We swallow it too fast, we choke. We get some in our lungs, we drown. However unreal it may seem, we are connected, you and I. We're on the same curve, just on opposite ends.

Normally, both your asses would be dead as fucking fried chicken, but you happen to pull this shit while I'm in a transitional period so I don't wanna kill you, I wanna help you. But I can't give you this case, it don't belong to me. Besides, I've already been through too much shit this morning over this case to hand it over to your dumb ass.

The path of the righteous man is beset on all sides by the iniquities of the selfish and the tyranny of evil men. Blessed is he who, in the name of charity and good will, shepherds the weak through the valley of darkness, for he is truly his brother's keeper and the finder of lost children. And I will strike down upon thee with great vengeance and furious anger those who would attempt to poison and destroy My brothers. And you will know My name is the Lord when I lay My vengeance upon thee.

My money's in that office, right? If she start giving me some bullshit about it ain't there, and we got to go someplace else and get it, I'm gonna shoot you in the head then and there. Then I'm gonna shoot that bitch in the kneecaps, find out where my goddamn money is. She gonna tell me too. Hey, look at me when I'm talking to you, motherfucker. You listen: we go in there, and that nigga Winston or anybody else is in there, you the first motherfucker to get shot. You understand?

Look, just because I don't be givin' no man a foot massage don't make it right for Marsellus to throw Antwone into a glass motherfuckin' house, fuckin' up the way the nigger talks. Motherfucker do that shit to me, he better paralyze my ass, 'cause I'll kill the motherfucker, know what I'm sayin'?

Well, the way they make shows is, they make one show. That show's called a pilot. Then they show that show to the people who make shows, and on the strength of that one show they decide if they're going to make more shows. Some pilots get picked and become television programs. Some don't, become nothing. She starred in one of the ones that became nothing.

Socket

OkHttp

```
HTTP/1.1 200 OK
Date: Thu, 9 Apr 2015 15:14:34 GMT
Server: JakeWharton/2.3.7 (Human)
Content-Type: application/presentation; charset=UTF-8
Content-Length: 3000
```

```
body.source();
```

```
HTTP/1.1 200 OK
Date: Thu, 9 Apr 2015 15:14:34 GMT
Server: JakeWharton/2.3.7 (Human)
Content-Type: application/presentation; charset=UTF-8
Content-Length: 3000
```

You think water moves fast? You should see ice. It moves like it has a mind. Like it knows it killed the world once and got a taste for murder. After the avalanche, it took us a week to climb out. Now, I don't know exactly when we turned on each other, but I know that seven of us survived the slide... and only five made it out. Now we took an oath, that I'm breaking now. We said we'd say it was the snow that killed the other two, but it wasn't. Nature is lethal but it doesn't hold a candle to man.

Your bones don't break, mine do. That's clear. Your cells react to bacteria and viruses differently than mine. You don't get sick, I do. That's also clear. But for some reason, you and I react the exact same way to water. We swallow it too fast, we choke. We get some in our lungs, we drown. However unreal it may seem, we are connected, you and I. We're on the same curve, just on opposite ends.

Normally, both your asses would be dead as fucking fried chicken, but you happen to pull this shit while I'm in a transitional period so I don't wanna kill you, I wanna help you. But I can't give you this case, it don't belong to me. Besides, I've already been through too much shit this morning over this case to hand it over to your dumb ass.

The path of the righteous man is beset on all sides by the iniquities of the selfish and the tyranny of evil men. Blessed is he who, in the name of charity and good will, shepherds the weak through the valley of darkness, for he is truly his brother's keeper and the finder of lost children. And I will strike down upon thee with great vengeance and furious anger those who would attempt to poison and destroy My brothers. And you will know My name is the Lord when I lay My vengeance upon thee.

My money's in that office, right? If she start giving me some bullshit about it ain't there, and we got to go someplace else and get it, I'm gonna shoot you in the head then and there. Then I'm gonna shoot that bitch in the kneecaps, find out where my goddamn money is. She gonna tell me too. Hey, look at me when I'm talking to you, motherfucker. You listen: we go in there, and that nigga Winston or anybody else is in there, you the first motherfucker to get shot. You understand?

Look, just because I don't be givin' no man a foot massage don't make it right for Marsellus to throw Antwone into a glass motherfuckin' house, fuckin' up the way the nigger talks. Motherfucker do that shit to me, he better paralyze my ass, 'cause I'll kill the motherfucker, know what I'm sayin'?

Well, the way they make shows is, they make one show. That show's called a pilot. Then they show that show to the people who make shows, and on the strength of that one show they decide if they're going to make more shows. Some pilots get picked and become television programs. Some don't, become nothing. She starred in one of the ones that became nothing.

Socket

OkHttp

HTTP/1.1 200 OK
Date: Thu, 9 Apr 2015 15:14:34 GMT
Server: JakeWharton/2.3.7 (Human)
Content-Type: application/presentation; charset=UTF-8
Content-Length: 3000

body.source();

FixedLengthSource

```
HTTP/1.1 200 OK
Date: Thu, 9 Apr 2015 15:14:34 GMT
Server: JakeWharton/2.3.7 (Human)
Content-Type: application/presentation; charset=UTF-8
Content-Length: 3000

You think water moves fast? You should see ice. It
moves like it has a mind. Like it knows it killed the
world once and got a taste for murder. After the
avalanche, it took us a week to climb out. Now, I don't
know exactly when we turned on each other, but I know
that seven of us survived the slide... and only five
made it out. Now we took an oath, that I'm breaking
now. We said we'd say it was the snow that killed the
other two, but it wasn't. Nature is lethal but it
doesn't hold a candle to man.

Your bones don't break, mine do. That's clear. Your
cells react to bacteria and viruses differently than
mine. You don't get sick, I do. That's also clear. But
for some reason, you and I react the exact same way to
water. We swallow it too fast, we choke. We get some in
our lungs, we drown. However unreal it may seem, we are
connected, you and I. We're on the same curve, just on
opposite ends.

Normally, both your asses would be dead as fucking
fried chicken, but you happen to pull this shit while
I'm in a transitional period so I don't wanna kill you,
I wanna help you. But I can't give you this case, it
don't belong to me. Besides, I've already been through
too much shit this morning over this case to hand it
over to your dumb ass.

The path of the righteous man is beset on all sides by
the iniquities of the selfish and the tyranny of evil
men. Blessed is he who, in the name of charity and good
will, shepherds the weak through the valley of
darkness, for he is truly his brother's keeper and the
finder of lost children. And I will strike down upon
thee with great vengeance and furious anger those who
would attempt to poison and destroy My brothers. And
you will know My name is the Lord when I lay My
vengeance upon thee.

My money's in that office, right? If she start giving
me some bullshit about it ain't there, and we got to go
someplace else and get it, I'm gonna shoot you in the
head then and there. Then I'm gonna shoot that bitch in
the kneecaps, find out where my goddamn money is. She
gonna tell me too. Hey, look at me when I'm talking to
you, motherfucker. You listen: we go in there, and that
nigga Winston or anybody else is in there, you the
first motherfucker to get shot. You understand?

Look, just because I don't be givin' no man a foot
massage don't make it right for Marsellus to throw
Antwone into a glass motherfuckin' house, fuckin' up
the way the nigger talks. Motherfucker do that shit to
me, he better paralyze my ass, 'cause I'll kill the
motherfucker, know what I'm sayin'?

Well, the way they make shows is, they make one show.
That show's called a pilot. Then they show that show to
the people who make shows, and on the strength of that
one show they decide if they're going to make more
shows. Some pilots get picked and become television
programs. Some don't, become nothing. She starred in
one of the ones that became nothing.
```

Socket

OkHttp

HTTP/1.1 200 OK
Date: Thu, 9 Apr 2015 15:14:34 GMT
Server: JakeWharton/2.3.7 (Human)
Content-Type: application/presentation; charset=UTF-8
Content-Length: 3000

body.source();

FixedLengthSource

```
HTTP/1.1 200 OK
Date: Thu, 9 Apr 2015 15:14:34 GMT
Server: JakeWharton/2.3.7 (Human)
Content-Type: application/presentation; charset=UTF-8
Content-Length: 3000

You think water moves fast? You should see ice. It
moves like it has a mind. Like it knows it killed the
world once and got a taste for murder. After the
avalanche, it took us a week to climb out. Now, I don't
know exactly when we turned on each other, but I know
that seven of us survived the slide... and only five
made it out. Now we took an oath, that I'm breaking
now. We said we'd say it was the snow that killed the
other two, but it wasn't. Nature is lethal but it
doesn't hold a candle to man.

Your bones don't break, mine do. That's clear. Your
cells react to bacteria and viruses differently than
mine. You don't get sick, I do. That's also clear. But
for some reason, you and I react the exact same way to
water. We swallow it too fast, we choke. We get some in
our lungs, we drown. However unreal it may seem, we are
connected, you and I. We're on the same curve, just on
opposite ends.

Normally, both your asses would be dead as fucking
fried chicken, but you happen to pull this shit while
I'm in a transitional period so I don't wanna kill you,
I wanna help you. But I can't give you this case, it
don't belong to me. Besides, I've already been through
too much shit this morning over this case to hand it
over to your dumb ass.

The path of the righteous man is beset on all sides by
the iniquities of the selfish and the tyranny of evil
men. Blessed is he who, in the name of charity and good
will, shepherds the weak through the valley of
darkness, for he is truly his brother's keeper and the
finder of lost children. And I will strike down upon
thee with great vengeance and furious anger those who
would attempt to poison and destroy My brothers. And
you will know My name is the Lord when I lay My
vengeance upon thee.

My money's in that office, right? If she start giving
me some bullshit about it ain't there, and we got to go
someplace else and get it, I'm gonna shoot you in the
head then and there. Then I'm gonna shoot that bitch in
the kneecaps, find out where my goddamn money is. She
gonna tell me too. Hey, look at me when I'm talking to
you, motherfucker. You listen: we go in there, and that
nigga Winston or anybody else is in there, you the
first motherfucker to get shot. You understand?

Look, just because I don't be givin' no man a foot
massage don't make it right for Marsellus to throw
Antwone into a glass motherfuckin' house, fuckin' up
the way the nigger talks. Motherfucker do that shit to
me, he better paralyze my ass, 'cause I'll kill the
motherfucker, know what I'm sayin'?

Well, the way they make shows is, they make one show.
That show's called a pilot. Then they show that show to
the people who make shows, and on the strength of that
one show they decide if they're going to make more
shows. Some pilots get picked and become television
programs. Some don't, become nothing. She starred in
one of the ones that became nothing.
```

Socket

OkHttp

HTTP/1.1 200 OK
Date: Thu, 9 Apr 2015 15:14:34 GMT
Server: JakeWharton/2.3.7 (Human)
Content-Type: application/presentation; charset=UTF-8
Content-Encoding: gzip
Content-Length: 3000

body.source();

GzipSource

FixedLengthSource

HTTP/1.1 200 OK
Date: Thu, 9 Apr 2015 15:14:34 GMT
Server: JakeWharton/2.3.7 (Human)
Content-Type: application/presentation; charset=UTF-8
Content-Encoding: gzip
Content-Length: 3000

You think water moves fast? You should see ice. It moves like it has a mind. Like it knows it killed the world once and got a taste for murder. After the avalanche, it took us a week to climb out. Now, I don't know exactly when we turned on each other, but I know that seven of us survived the slide... and only five made it out. Now we took an oath, that I'm breaking now. We said we'd say it was the snow that killed the other two, but it wasn't. Nature is lethal but it doesn't hold a candle to man.

Your bones don't break, mine do. That's clear. Your cells react to bacteria and viruses differently than mine. You don't get sick, I do. That's also clear. But for some reason, you and I react the exact same way to water. We swallow it too fast, we choke. We get some in our lungs, we drown. However unreal it may seem, we are connected, you and I. We're on the same curve, just on opposite ends.

Normally, both your asses would be dead as fucking fried chicken, but you happen to pull this shit while I'm in a transitional period so I don't wanna kill you, I wanna help you. But I can't give you this case, it don't belong to me. Besides, I've already been through too much shit this morning over this case to hand it over to your dumb ass.

The path of the righteous man is beset on all sides by the iniquities of the selfish and the tyranny of evil men. Blessed is he who, in the name of charity and good will, shepherds the weak through the valley of darkness, for he is truly his brother's keeper and the finder of lost children. And I will strike down upon thee with great vengeance and furious anger those who would attempt to poison and destroy My brothers. And you will know My name is the Lord when I lay My vengeance upon thee.

My money's in that office, right? If she start giving me some bullshit about it ain't there, and we got to go someplace else and get it, I'm gonna shoot you in the head then and there. Then I'm gonna shoot that bitch in the kneecaps, find out where my goddamn money is. She gonna tell me too. Hey, look at me when I'm talking to you, motherfucker. You listen: we go in there, and that nigga Winston or anybody else is in there, you the first motherfucker to get shot. You understand?

Look, just because I don't be givin' no man a foot massage don't make it right for Marsellus to throw Antwone into a glass motherfuckin' house, fuckin' up the way the nigger talks. Motherfucker do that shit to me, he better paralyze my ass, 'cause I'll kill the motherfucker, know what I'm sayin'?

Well, the way they make shows is, they make one show. That show's called a pilot. Then they show that show to the people who make shows, and on the strength of that one show they decide if they're going to make more shows. Some pilots get picked and become television programs. Some don't, become nothing. She starred in one of the ones that became nothing.

Socket

OkHttp

HTTP/1.1 200 OK
Date: Thu, 9 Apr 2015 15:14:34 GMT
Server: JakeWharton/2.3.7 (Human)
Content-Type: application/presentation; charset=UTF-8
Transfer-Encoding: **chunked**

`body.source();`

ChunkedSource

```
HTTP/1.1 200 OK
Date: Thu, 9 Apr 2015 15:14:34 GMT
Server: JakeWharton/2.3.7 (Human)
Content-Type: application/presentation; charset=UTF-8
Transfer-Encoding: chunked

1F6
You think water moves fast? You should see ice. It
moves like it has a mind. Like it knows it killed the
world once and got a taste for murder. After the
avalanche, it took us a week to climb out. Now, I don't
know exactly when we turned on each other, but I know
that seven of us survived the slide... and only five
made it out. Now we took an oath, that I'm breaking
now. We said we'd say it was the snow that killed the
other two, but it wasn't. Nature is lethal but it
doesn't hold a candle to man.

18C
Your bones don't break, mine do. That's clear. Your
cells react to bacteria and viruses differently than
mine. You don't get sick, I do. That's also clear. But
for some reason, you and I react the exact same way to
water. We swallow it too fast, we choke. We get some in
our lungs, we drown. However unreal it may seem, we are
connected, you and I. We're on the same curve, just on
opposite ends.

158
Normally, both your asses would be dead as fucking
fried chicken, but you happen to pull this shit while
I'm in a transitional period so I don't wanna kill you,
I wanna help you. But I can't give you this case, it
don't belong to me. Besides, I've already been through
too much shit this morning over this case to hand it
over to your dumb ass.

1EF
The path of the righteous man is beset on all sides by
the iniquities of the selfish and the tyranny of evil
men. Blessed is he who, in the name of charity and good
will, shepherds the weak through the valley of
darkness, for he is truly his brother's keeper and the
finder of lost children. And I will strike down upon
thee with great vengeance and furious anger those who
would attempt to poison and destroy My brothers. And
you will know My name is the Lord when I lay My
vengeance upon thee.

1E3
My money's in that office, right? If she start giving
me some bullshit about it ain't there, and we got to go
someplace else and get it, I'm gonna shoot you in the
head then and there. Then I'm gonna shoot that bitch in
the kneecaps, find out where my goddamn money is. She
gonna tell me too. Hey, look at me when I'm talking to
you, motherfucker. You listen: we go in there, and that
nigga Winston or anybody else is in there, you the
first motherfucker to get shot. You understand?

12B
Look, just because I don't be givin' no man a foot
massage don't make it right for Marsellus to throw
Antwone into a glass motherfuckin' house, fuckin' up
the way the nigger talks. Motherfucker do that shit to
me, he better paralyze my ass, 'cause I'll kill the
motherfucker, know what I'm sayin'?

165
Well, the way they make shows is, they make one show.
That show's called a pilot. Then they show that show to
the people who make shows, and on the strength of that
one show they decide if they're going to make more
shows. Some pilots get picked and become television
programs. Some don't, become nothing. She starred in
one of the ones that became nothing.
```

Socket

OkHttp

HTTP/1.1 200 OK
Date: Thu, 9 Apr 2015 15:14:34 GMT
Server: JakeWharton/2.3.7 (Human)
Content-Type: application/presentation; charset=UTF-8
Transfer-Encoding: **chunked**

body.source();

ChunkedSource

```
HTTP/1.1 200 OK
Date: Thu, 9 Apr 2015 15:14:34 GMT
Server: JakeWharton/2.3.7 (Human)
Content-Type: application/presentation; charset=UTF-8
Transfer-Encoding: chunked

1F6
You think water moves fast? You should see ice. It
moves like it has a mind. Like it knows it killed the
world once and got a taste for murder. After the
avalanche, it took us a week to climb out. Now, I don't
know exactly when we turned on each other, but I know
that seven of us survived the slide... and only five
made it out. Now we took an oath, that I'm breaking
now. We said we'd say it was the snow that killed the
other two, but it wasn't. Nature is lethal but it
doesn't hold a candle to man.

18C
Your bones don't break, mine do. That's clear. Your
cells react to bacteria and viruses differently than
mine. You don't get sick, I do. That's also clear. But
for some reason, you and I react the exact same way to
water. We swallow it too fast, we choke. We get some in
our lungs, we drown. However unreal it may seem, we are
connected, you and I. We're on the same curve, just on
opposite ends.

158
Normally, both your asses would be dead as fucking
fried chicken, but you happen to pull this shit while
I'm in a transitional period so I don't wanna kill you,
I wanna help you. But I can't give you this case, it
don't belong to me. Besides, I've already been through
too much shit this morning over this case to hand it
over to your dumb ass.

1EF
The path of the righteous man is beset on all sides by
the iniquities of the selfish and the tyranny of evil
men. Blessed is he who, in the name of charity and good
will, shepherds the weak through the valley of
darkness, for he is truly his brother's keeper and the
finder of lost children. And I will strike down upon
thee with great vengeance and furious anger those who
would attempt to poison and destroy My brothers. And
you will know My name is the Lord when I lay My
vengeance upon thee.

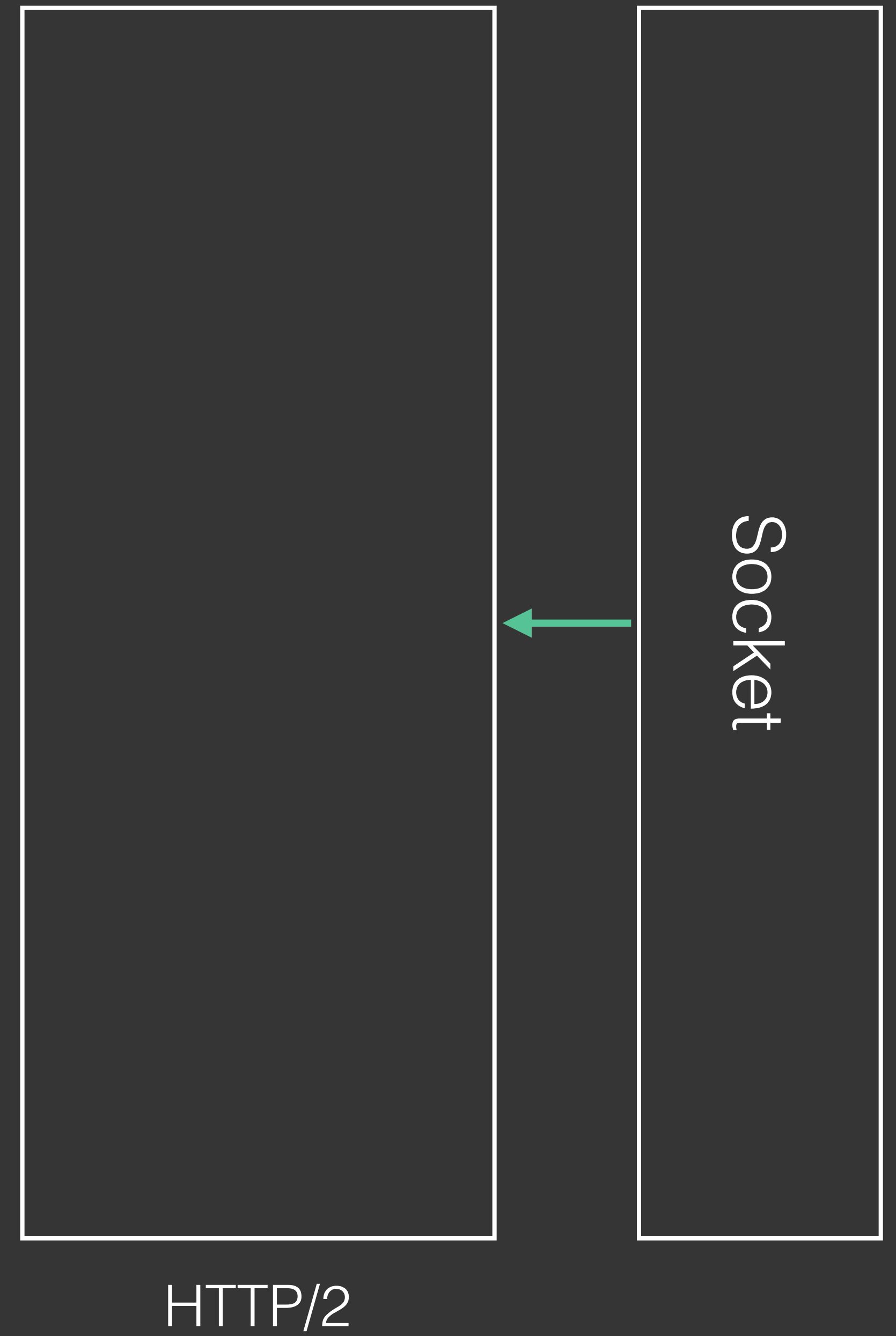
1E3
My money's in that office, right? If she start giving
me some bullshit about it ain't there, and we got to go
someplace else and get it, I'm gonna shoot you in the
head then and there. Then I'm gonna shoot that bitch in
the kneecaps, find out where my goddamn money is. She
gonna tell me too. Hey, look at me when I'm talking to
you, motherfucker. You listen: we go in there, and that
nigga Winston or anybody else is in there, you the
first motherfucker to get shot. You understand?

12B
Look, just because I don't be givin' no man a foot
massage don't make it right for Marsellus to throw
Antwone into a glass motherfuckin' house, fuckin' up
the way the nigger talks. Motherfucker do that shit to
me, he better paralyze my ass, 'cause I'll kill the
motherfucker, know what I'm sayin'?

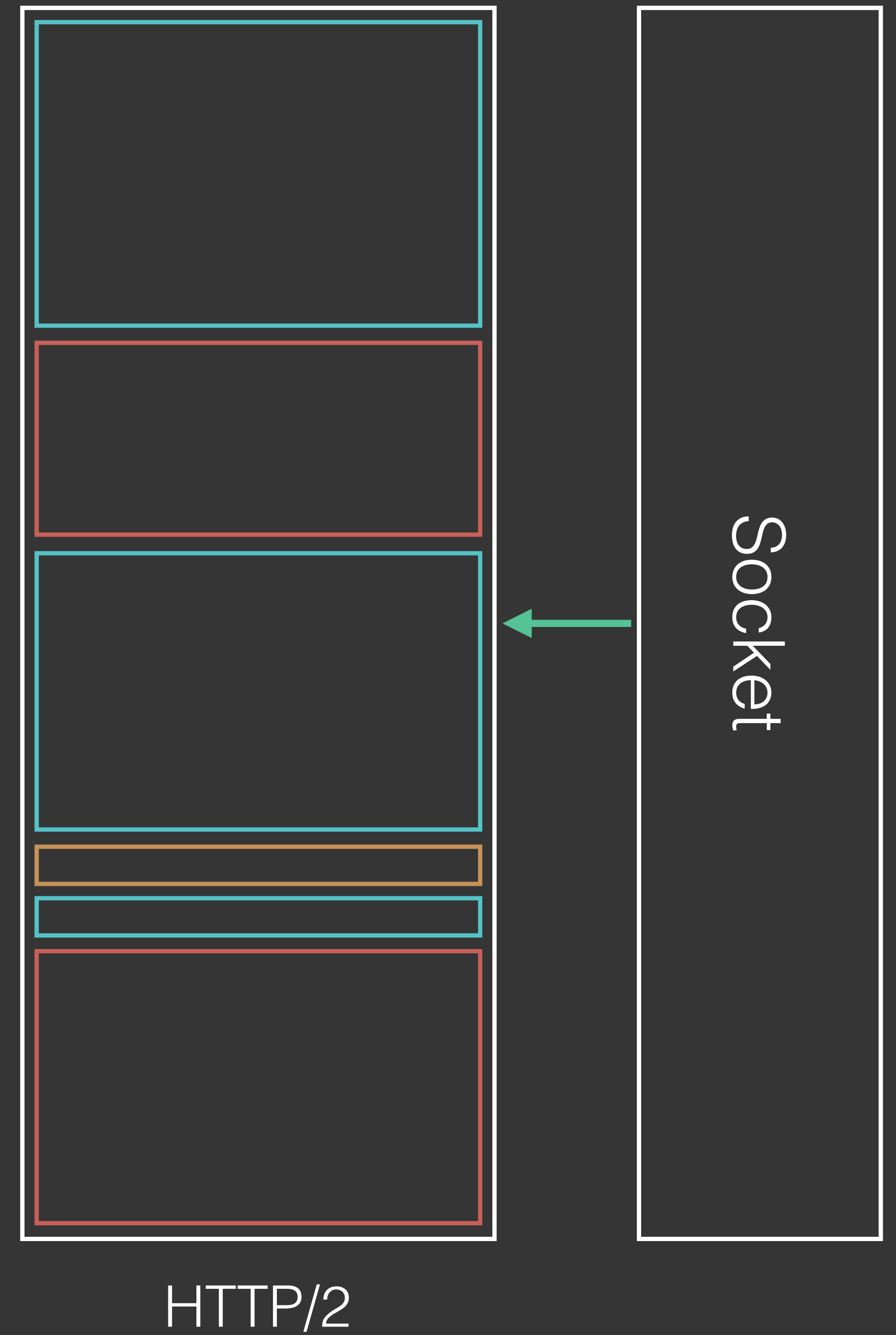
165
Well, the way they make shows is, they make one show.
That show's called a pilot. Then they show that show to
the people who make shows, and on the strength of that
one show they decide if they're going to make more
shows. Some pilots get picked and become television
programs. Some don't, become nothing. She starred in
one of the ones that became nothing.
```

Socket

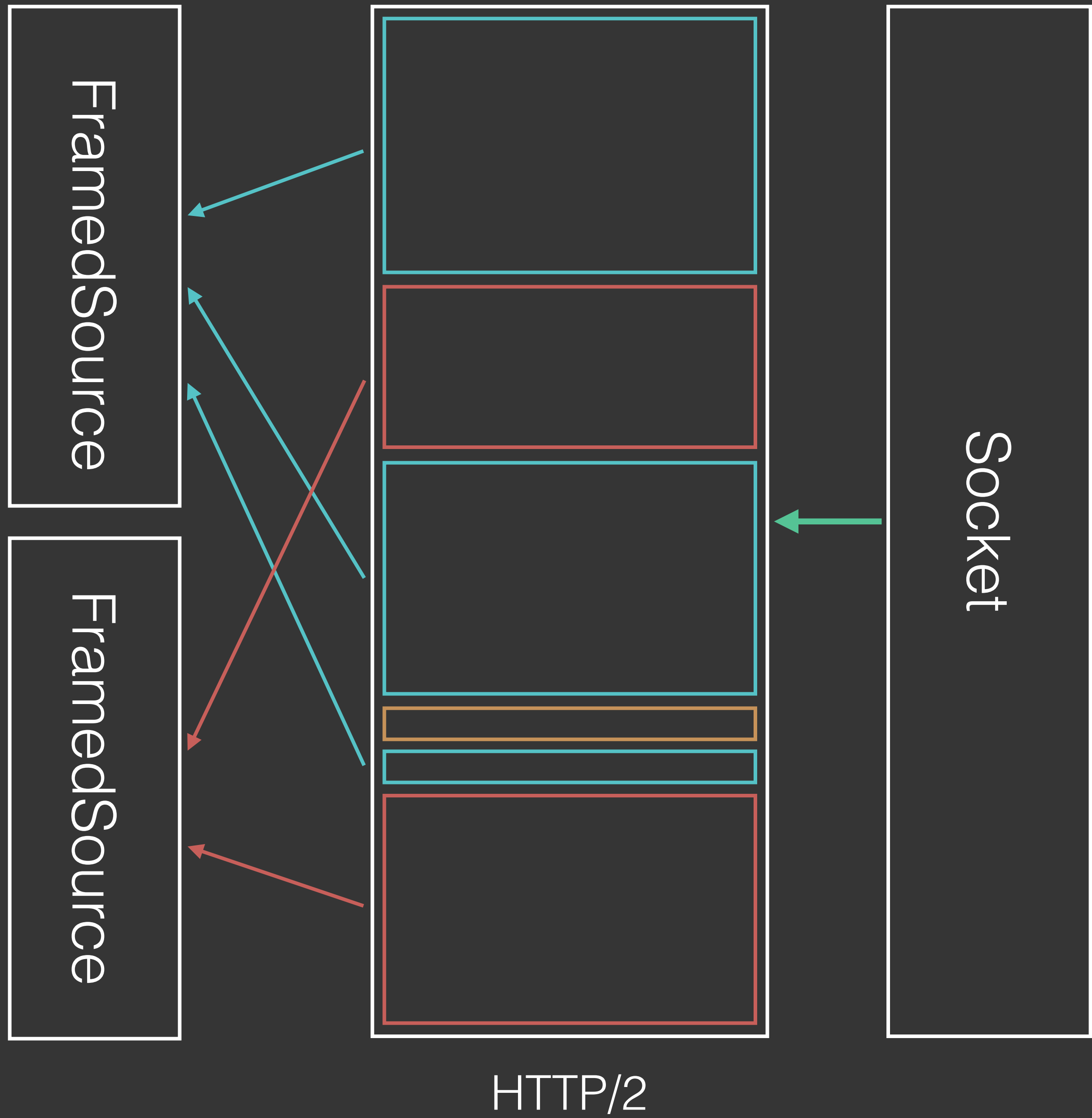
OkHttp



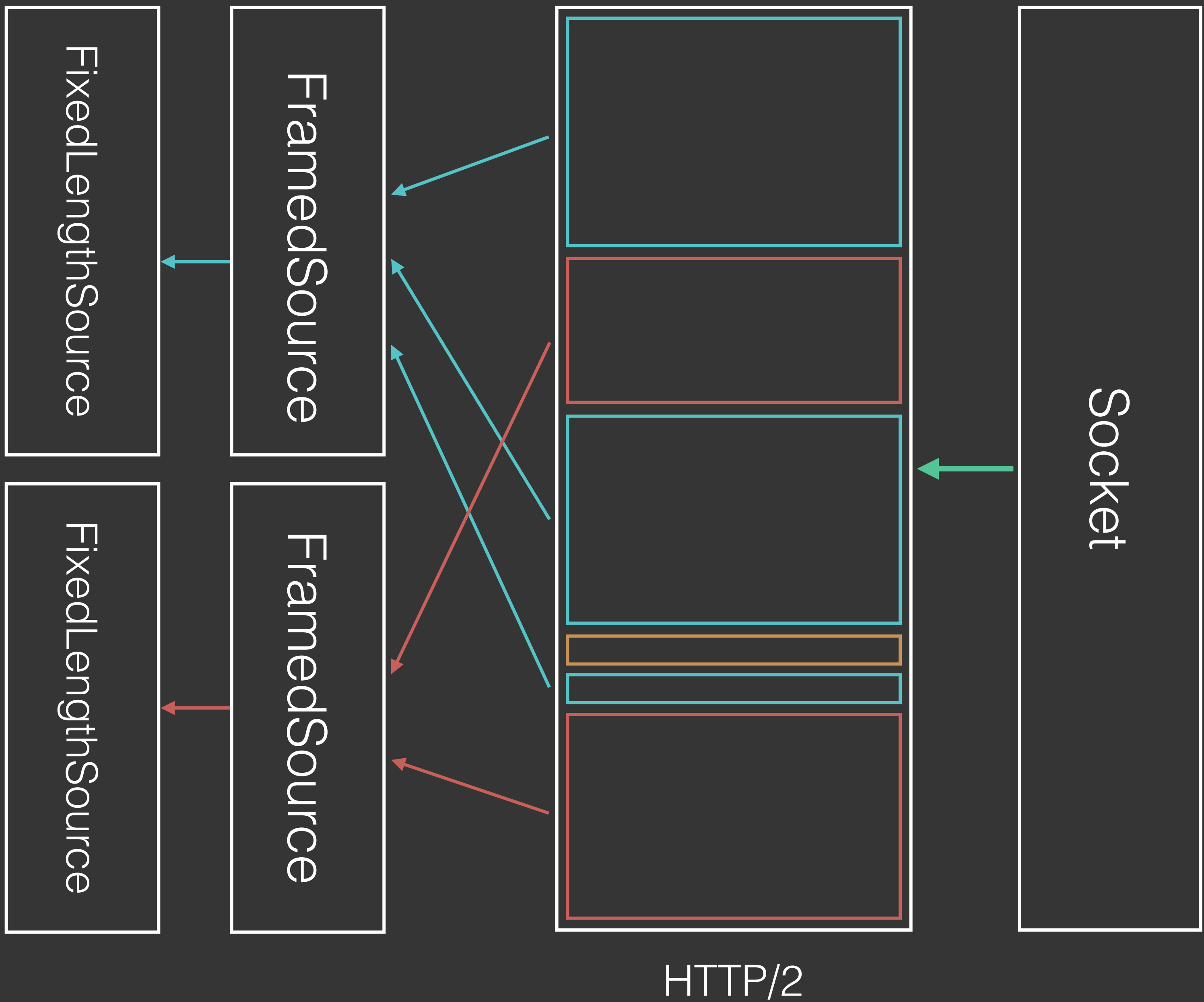
OkHttp



OkHttp



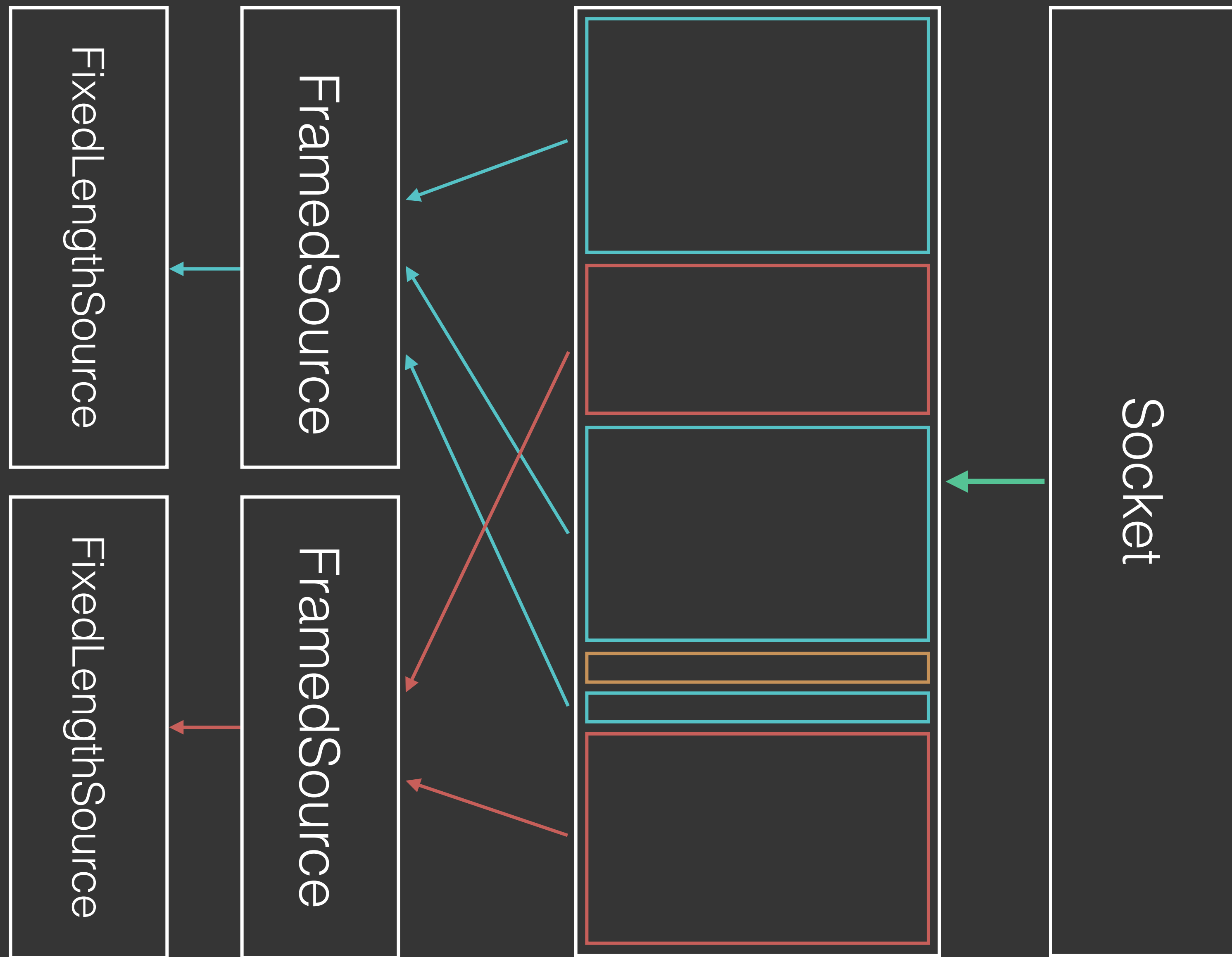
OkHttp



OkHttp

`body1.source();`

`body2.source();`



HTTP/2

Moshi

Moshi

- `JsonReader` / `JsonWriter` for streaming

Moshi

```
Sink sink = //...  
JsonWriter writer = new JsonWriter(sink);
```


Moshi

```
Sink sink = //...  
JsonWriter writer = new JsonWriter(sink);  
  
writer.beginObject();
```

```
{
```

Moshi

```
Sink sink = //...  
JsonWriter writer = new JsonWriter(sink);
```

```
writer.beginObject();  
writer.name("title")  
    .value("A Few 'Ok' Libraries");
```

```
{  
    "title": "A Few 'Ok' Libraries"
```

Moshi

```
Sink sink = //...  
JsonWriter writer = new JsonWriter(sink);
```

```
writer.beginObject();  
writer.name("title")  
    .value("A Few 'Ok' Libraries");  
writer.name("presenter")  
    .value("Jake Wharton");
```

```
{  
    "title": "A Few 'Ok' Libraries",  
    "presenter": "Jake Wharton"
```

Moshi

```
Sink sink = //...  
JsonWriter writer = new JsonWriter(sink);
```

```
writer.beginObject();  
writer.name("title")  
    .value("A Few 'Ok' Libraries");  
writer.name("presenter")  
    .value("Jake Wharton");  
writer.name("tags")  
    .beginArray();
```

```
{  
    "title": "A Few 'Ok' Libraries",  
    "presenter": "Jake Wharton",  
    "tags": [  
        "tag1",  
        "tag2",  
        "tag3"  
    ]  
}
```

Moshi

```
Sink sink = //...  
JsonWriter writer = new JsonWriter(sink);
```

```
writer.beginObject();  
writer.name("title")  
    .value("A Few 'Ok' Libraries");  
writer.name("presenter")  
    .value("Jake Wharton");  
writer.name("tags")  
    .beginArray()  
    .value("Okio")  
    .value("OkHttp")  
    .value("Moshi")  
    .value("Retrofit");
```

```
{  
  "title": "A Few 'Ok' Libraries",  
  "presenter": "Jake Wharton",  
  "tags": [  
    "Okio",  
    "OkHttp",  
    "Moshi",  
    "Retrofit"
```

Moshi

```
Sink sink = //...  
JsonWriter writer = new JsonWriter(sink);
```

```
writer.beginObject();  
writer.name("title")  
    .value("A Few 'Ok' Libraries");  
writer.name("presenter")  
    .value("Jake Wharton");  
writer.name("tags")  
    .beginArray()  
    .value("Okio")  
    .value("OkHttp")  
    .value("Moshi")  
    .value("Retrofit")  
    .endArray();
```

```
{  
  "title": "A Few 'Ok' Libraries",  
  "presenter": "Jake Wharton",  
  "tags": [  
    "Okio",  
    "OkHttp",  
    "Moshi",  
    "Retrofit"  
  ]  
}
```

Moshi

```
Sink sink = //...  
JsonWriter writer = new JsonWriter(sink);
```

```
writer.beginObject();  
writer.name("title")  
    .value("A Few 'Ok' Libraries");  
writer.name("presenter")  
    .value("Jake Wharton");  
writer.name("tags")  
    .beginArray()  
    .value("Okio")  
    .value("OkHttp")  
    .value("Moshi")  
    .value("Retrofit")  
    .endArray();  
writer.endObject();
```

```
{  
  "title": "A Few 'Ok' Libraries",  
  "presenter": "Jake Wharton",  
  "tags": [  
    "Okio",  
    "OkHttp",  
    "Moshi",  
    "Retrofit"  
  ]  
}
```

Moshi

```
Sink sink = //...  
JsonWriter writer = new JsonWriter(sink);
```

```
writer.beginObject();  
writer.name("title")  
    .value("A Few 'Ok' Libraries");  
writer.name("presenter")  
    .value("Jake Wharton");  
writer.name("tags")  
    .beginArray()  
    .value("Okio")  
    .value("OkHttp")  
    .value("Moshi")  
    .value("Retrofit")  
    .endArray();  
writer.endObject();
```

```
writer.close();
```

```
{  
  "title": "A Few 'Ok' Libraries",  
  "presenter": "Jake Wharton",  
  "tags": [  
    "Okio",  
    "OkHttp",  
    "Moshi",  
    "Retrofit"  
  ]  
}
```


Moshi

```
Sink sink = //...
JsonWriter writer = new JsonWriter(sink);

writer.beginObject();
writer.name("title")
    .value("A Few 'Ok' Libraries");
writer.name("presenter")
    .value("Jake Wharton");
writer.name("tags")
    .beginArray()
    .value("Okio")
    .value("OkHttp")
    .value("Moshi")
    .value("Retrofit")
    .endArray();
writer.endObject();

writer.close();
```



```
Sink sink = //...
BufferedSink s = Okio.buffer(sink);

s.writeByte('{');
s.writeByte('').writeUtf8("title").writeByte('');
s.writeByte(':').writeByte('')
    .writeUtf8("A Few 'Ok' Libraries")
    .writeByte('');
s.writeByte(',');
s.writeByte('').writeUtf8("presenter").writeByte('');
s.writeByte(':').writeByte('')
    .writeUtf8("Jake Wharton")
    .writeByte('');
s.writeByte(',');
s.writeByte('').writeUtf8("tags").writeByte('');
s.writeByte(':').writeByte('[');
s.writeByte('').writeUtf8("Okio").writeByte('');
s.writeByte(',');
s.writeByte('').writeUtf8("OkHttp").writeByte('');
s.writeByte(',');
s.writeByte('').writeUtf8("Moshi").writeByte('');
s.writeByte(',');
s.writeByte('').writeUtf8("Retrofit").writeByte('');
s.writeByte(']');
s.writeByte('}');

s.close();
```

Moshi

```
Sink sink = //...  
JsonWriter writer = new JsonWriter(sink);
```

```
writer.beginObject();  
writer.name("title")  
    .value("A Few 'Ok' Libraries");  
writer.name("presenter")  
    .value("Jake Wharton");  
writer.name("tags")  
    .beginArray()  
    .value("Okio")  
    .value("OkHttp")  
    .value("Moshi")  
    .value("Retrofit")  
    .endArray();  
writer.endObject();
```

```
writer.close();
```

```
{  
  "title": "A Few 'Ok' Libraries",  
  "presenter": "Jake Wharton",  
  "tags": [  
    "Okio",  
    "OkHttp",  
    "Moshi",  
    "Retrofit"  
  ]  
}
```

Moshi

```
Source source = //...  
JsonReader reader = new JsonReader(source);
```

```
{  
    "title": "A Few 'Ok' Libraries",  
    "presenter": "Jake Wharton",  
    "tags": [  
        "Okio",  
        "OkHttp",  
        "Moshi",  
        "Retrofit"  
    ]  
}
```

Moshi

```
Source source = //...
JsonReader reader = new JsonReader(source);


reader.beginObject();
while (reader.peek() != Token.END_OBJECT) {
    switch (reader.nextName()) {
        case "title":
            String title = reader.nextString();
            // TODO do something with 'title'...
            break;
        case "presenter":
            String presenter = reader.nextString();
            // TODO do something with 'presenter'...
            break;
        case "tags":
            reader.beginArray();
            Set<String> tags = new LinkedHashSet<>();
            while (reader.peek() != Token.END_ARRAY) {
                tags.add(reader.nextString());
            }
            reader.endArray();
            // TODO do something with 'tags'...
            break;
        default: break; // Ignored
    }
}
reader.endObject();
reader.close();
```

```
{
    "title": "A Few 'Ok' Libraries",
    "presenter": "Jake Wharton",
    "tags": [
        "Okio",
        "OkHttp",
        "Moshi",
        "Retrofit"
    ]
}
```

Moshi

```
Source source = //...
JsonReader reader = new JsonReader(source);

reader.beginObject();
while (reader.peek() != Token.END_OBJECT) {
    switch (reader.nextName()) {
        case "title":
            String title = reader.nextString();
            // TODO do something with 'title'...
            break;
        case "presenter":
            String presenter = reader.nextString();
            // TODO do something with 'presenter'...
            break;
        case "tags":
            reader.beginArray();
            Set<String> tags = new LinkedHashSet<>();
            while (reader.peek() != Token.END_ARRAY) {
                tags.add(reader.nextString());
            }
            reader.endArray();
            // TODO do something with 'tags'...
            break;
        default: break; // Ignored
    }
}
reader.endObject();
reader.close();
```



```
static final ByteString DOUBLE_QUOTE =
    ByteString.encodeUtf8("\"");

long i = s.indexOfElement(DOUBLE_QUOTE);
String string = s.readUtf8(i);
```

Moshi

- `JsonReader` / `JsonWriter` for streaming

Moshi

- `JsonReader` / `JsonWriter` for streaming
- `JsonAdapter` for object mapping

Moshi

```
{  
  "title": "A Few 'Ok' Libraries",  
  "presenter": "Jake Wharton",  
  "tags": [  
    "Okio",  
    "OkHttp",  
    "Moshi",  
    "Retrofit"  
  ]  
}
```


Moshi

```
class Presentation {  
    String title;  
    String presenter;  
    Set<String> tags;  
}
```

```
{  
    "title": "A Few 'Ok' Libraries",  
    "presenter": "Jake Wharton",  
    "tags": [  
        "Okio",  
        "OkHttp",  
        "Moshi",  
        "Retrofit"  
    ]  
}
```

Moshi

```
class Presentation {  
    String title;  
    String presenter;  
    Set<String> tags;  
}
```

```
JsonAdapter<Presentation> adapter  
    = moshi.adapter(Presentation.class);
```

```
Presentation presentation =  
    adapter.fromJson(source);
```

```
{  
    "title": "A Few 'Ok' Libraries",  
    "presenter": "Jake Wharton",  
    "tags": [  
        "Okio",  
        "OkHttp",  
        "Moshi",  
        "Retrofit"  
    ]  
}
```

Retrofit

Retrofit

```
interface DroidconMtl {  
    @GET("/a-few-ok-libraries.json")  
    Presentation okLibraries();  
}
```

Retrofit

```
interface DroidconMtl {  
    @GET("/a-few-ok-libraries.json")  
    Presentation okLibraries();  
}
```

```
class Presentation {  
    String title;  
    String presenter;  
    Set<String> tags;  
}
```

Retrofit

```
interface DroidconMtl {  
    @GET("/a-few-ok-libraries.json")  
    Presentation okLibraries();  
}
```

```
RestAdapter ra = new RestAdapter.Builder()  
    .setEndpoint("http://droidcon.ca")  
    .build();
```

Retrofit

```
interface DroidconMtl {  
    @GET("/a-few-ok-libraries.json")  
    Presentation okLibraries();  
}
```

```
RestAdapter ra = new RestAdapter.Builder()  
    .setEndpoint("http://droidcon.ca")  
    .build();
```

```
DroidconMtl droidconMtl = ra.create(DroidconMtl.class);
```

Retrofit

```
interface DroidconMtl {  
    @GET("/a-few-ok-libraries.json")  
    Presentation okLibraries();  
}
```

```
RestAdapter ra = new RestAdapter.Builder()  
    .setEndpoint("http://droidcon.ca")  
    .build();
```

```
DroidconMtl droidconMtl = ra.create(DroidconMtl.class);
```

```
Presentation ok = droidconMtl.okLibraries();
```


Retrofit

```
RestAdapter ra = new RestAdapter.Builder()  
    .setEndpoint("http://droidcon.ca")  
    .build();
```

Retrofit

```
RestAdapter ra = new RestAdapter.Builder()  
    .setEndpoint("http://droidcon.ca")  
    .setClient(new OkHttpClient())  
    .build();
```

Retrofit

```
RestAdapter ra = new RestAdapter.Builder()  
    .setEndpoint("http://droidcon.ca")  
    .setClient(new OkHttpClient())  
    .setConverter(new MoshiConverter())  
    .build();
```

Retrofit

Retrofit

OkHttp

Retrofit

OkHttp

Socket

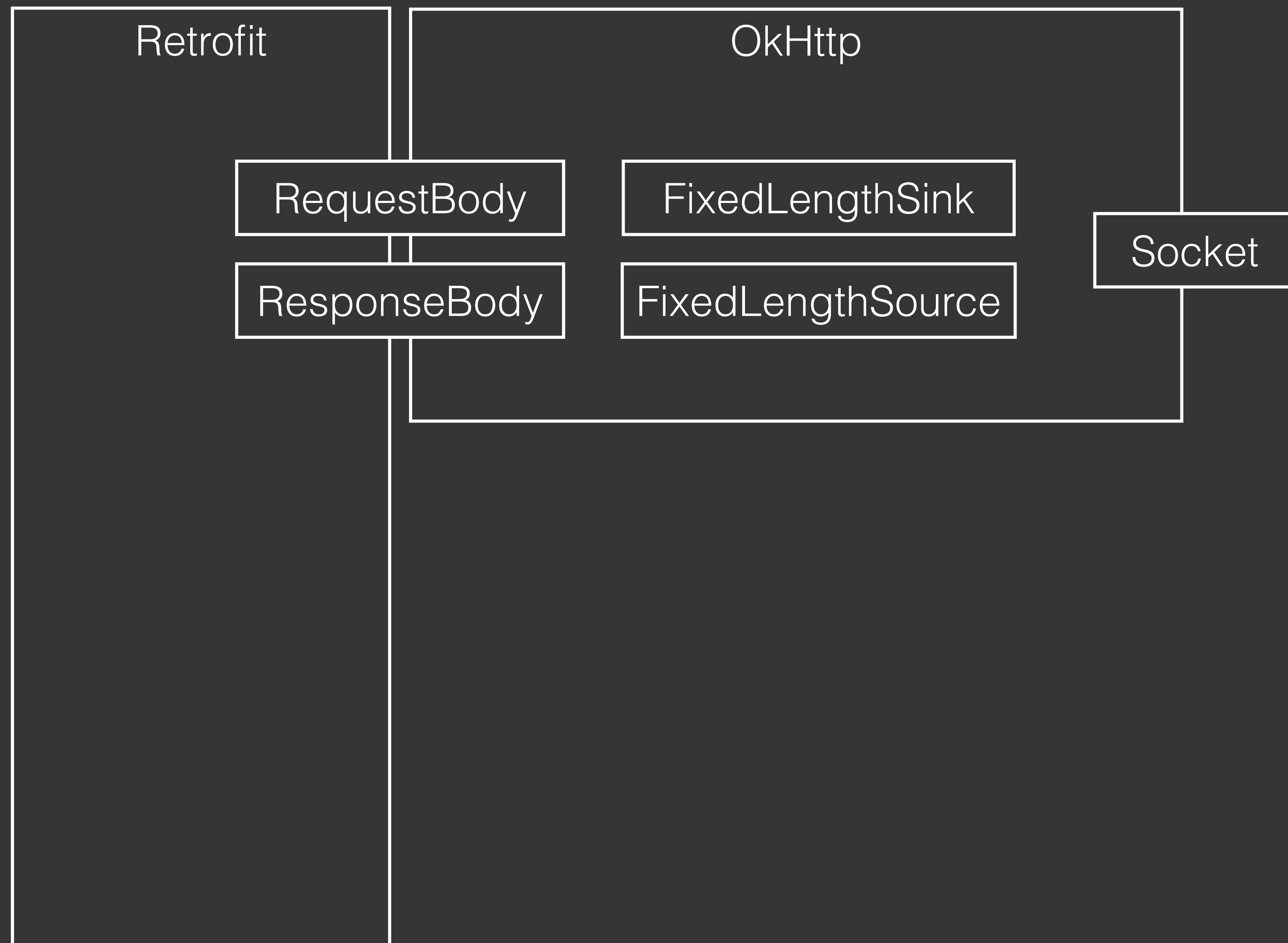
Retrofit

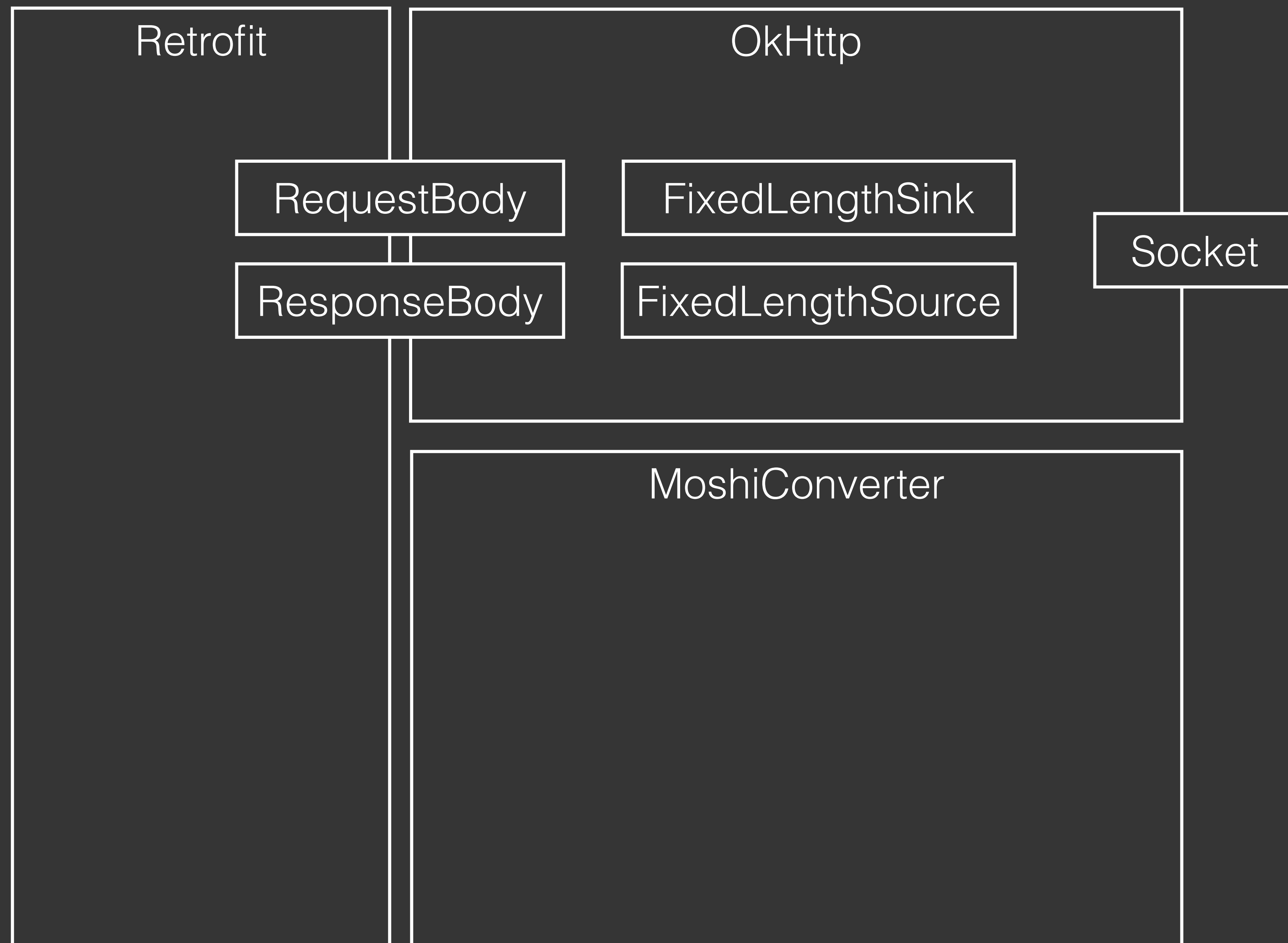
OkHttp

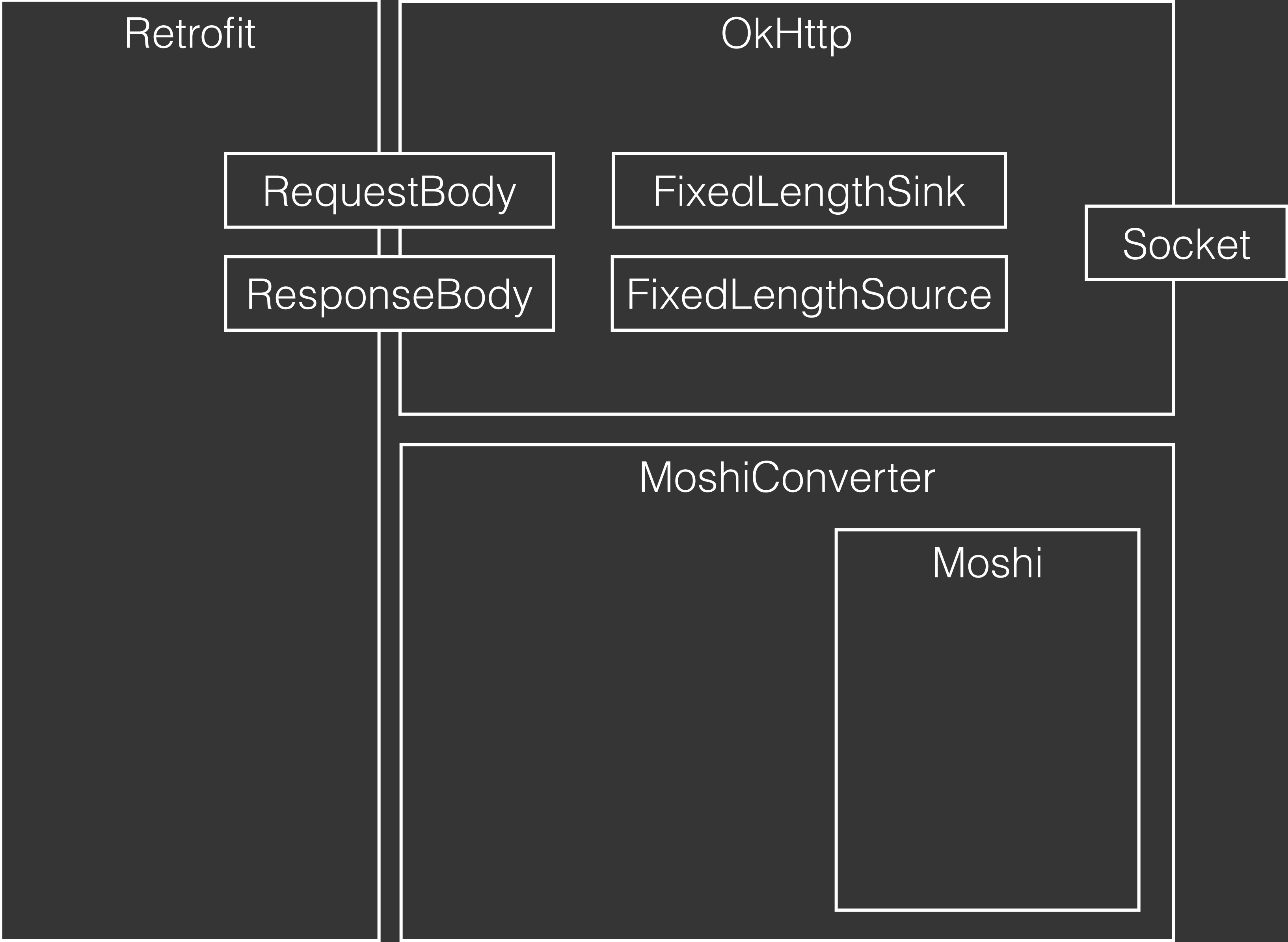
FixedLengthSink

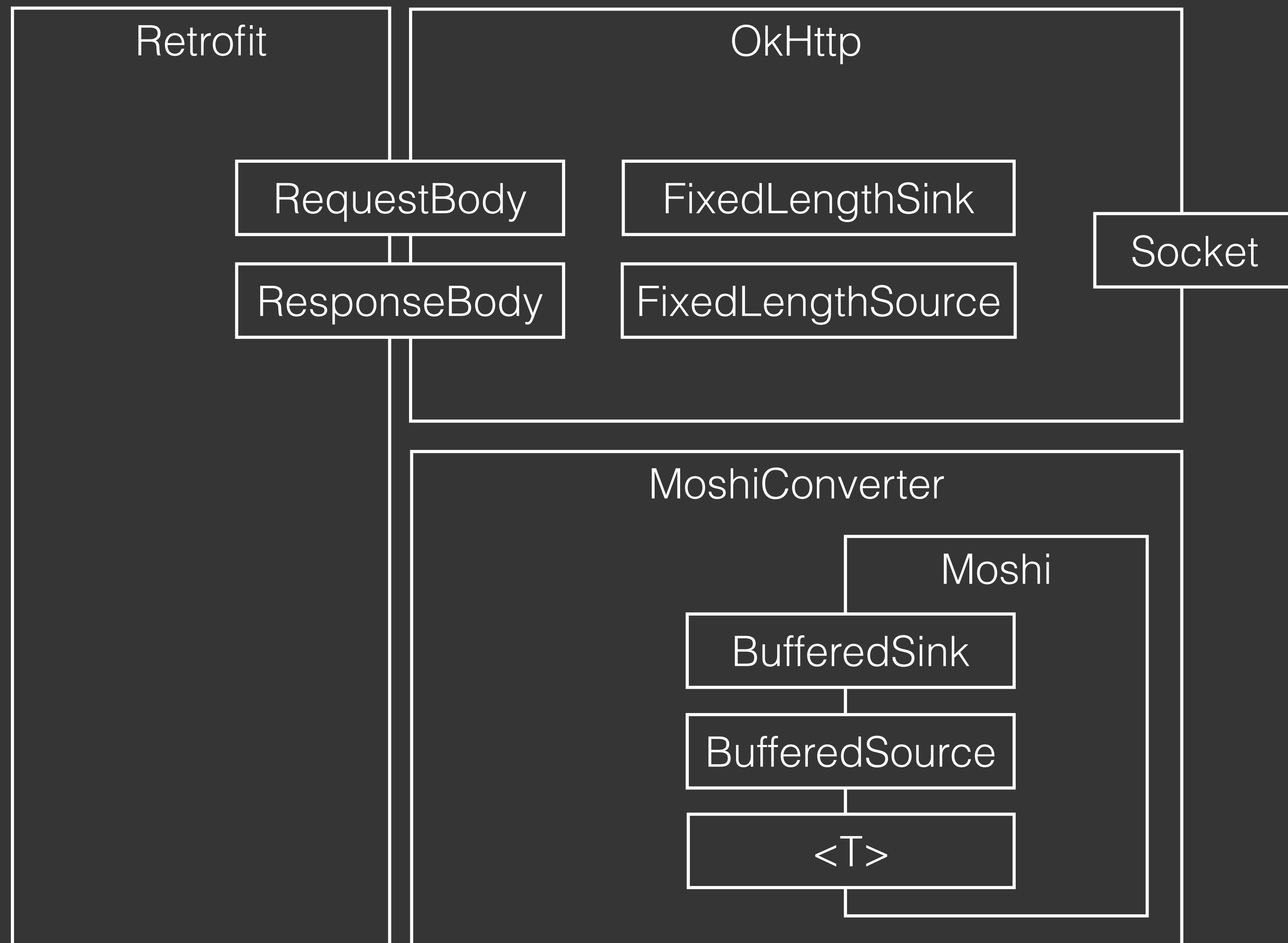
FixedLengthSource

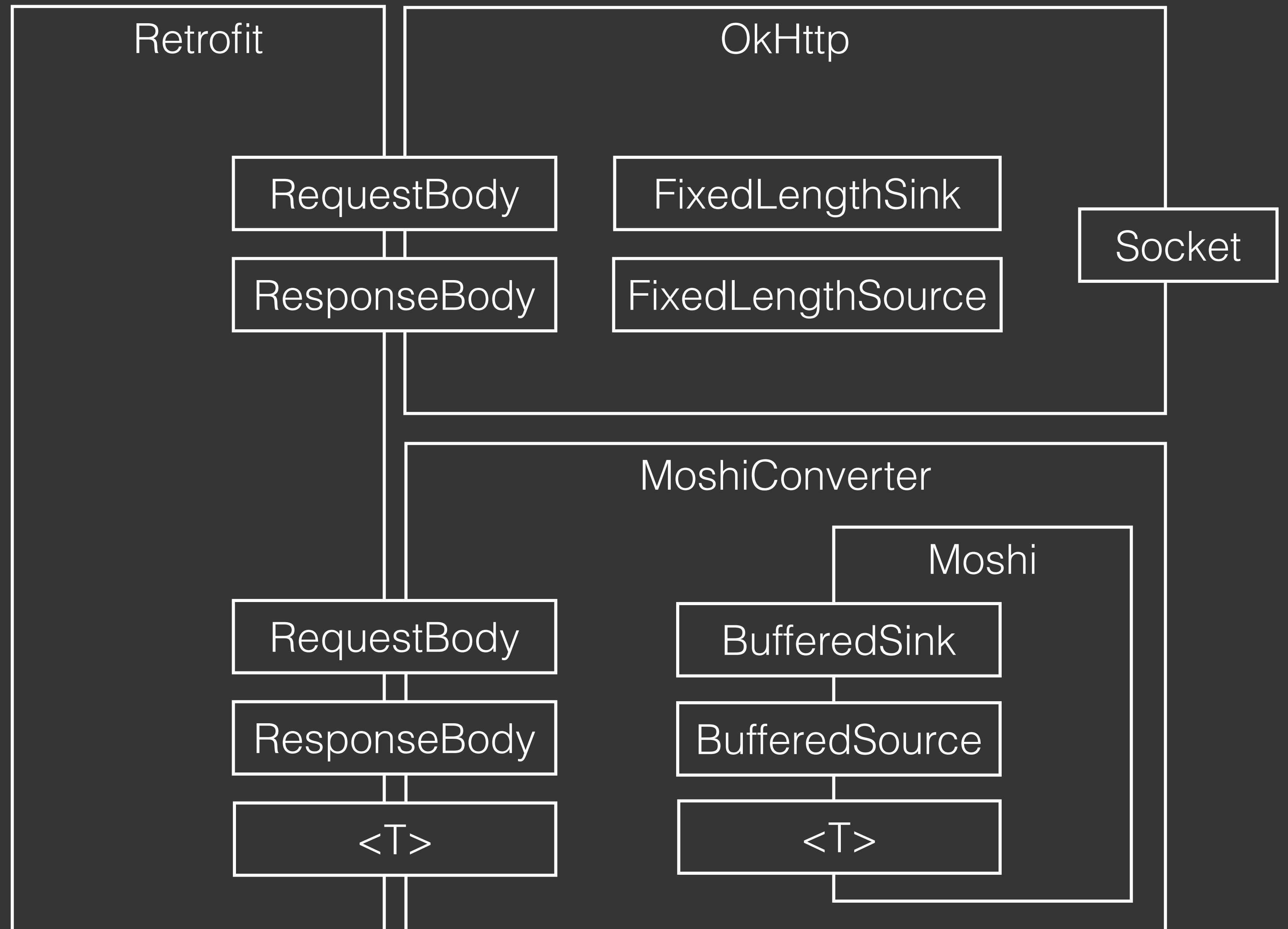
Socket

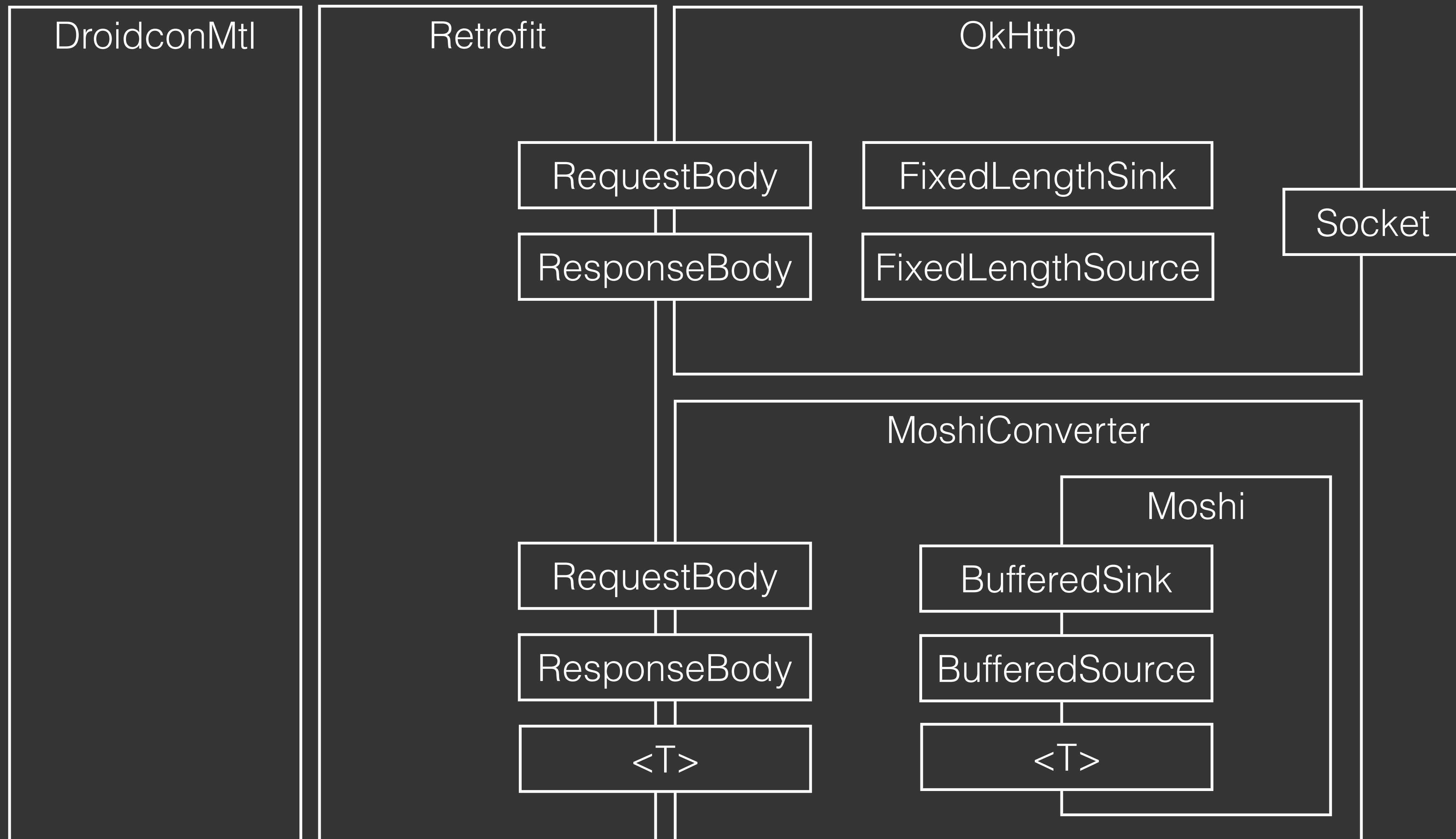


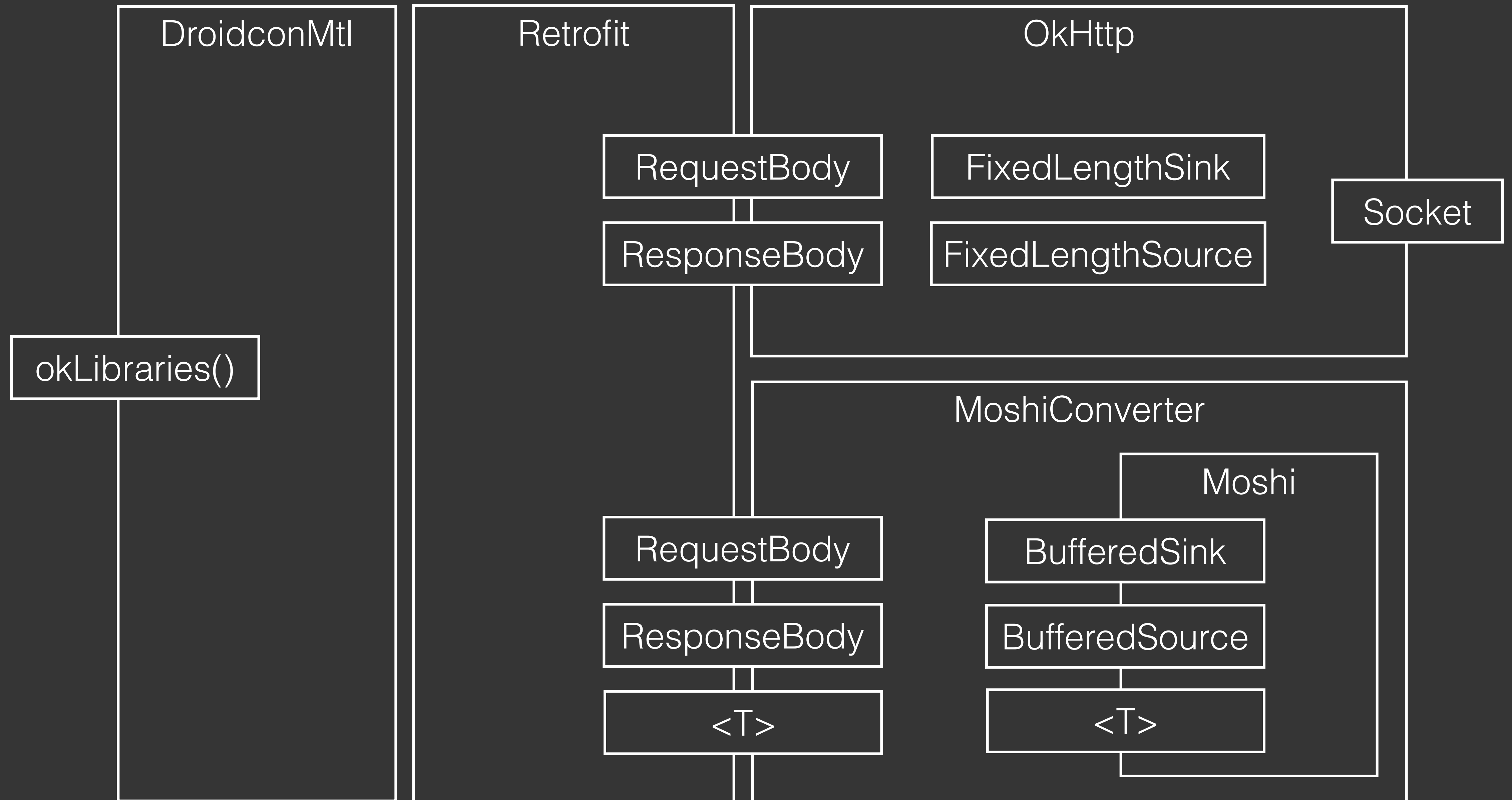


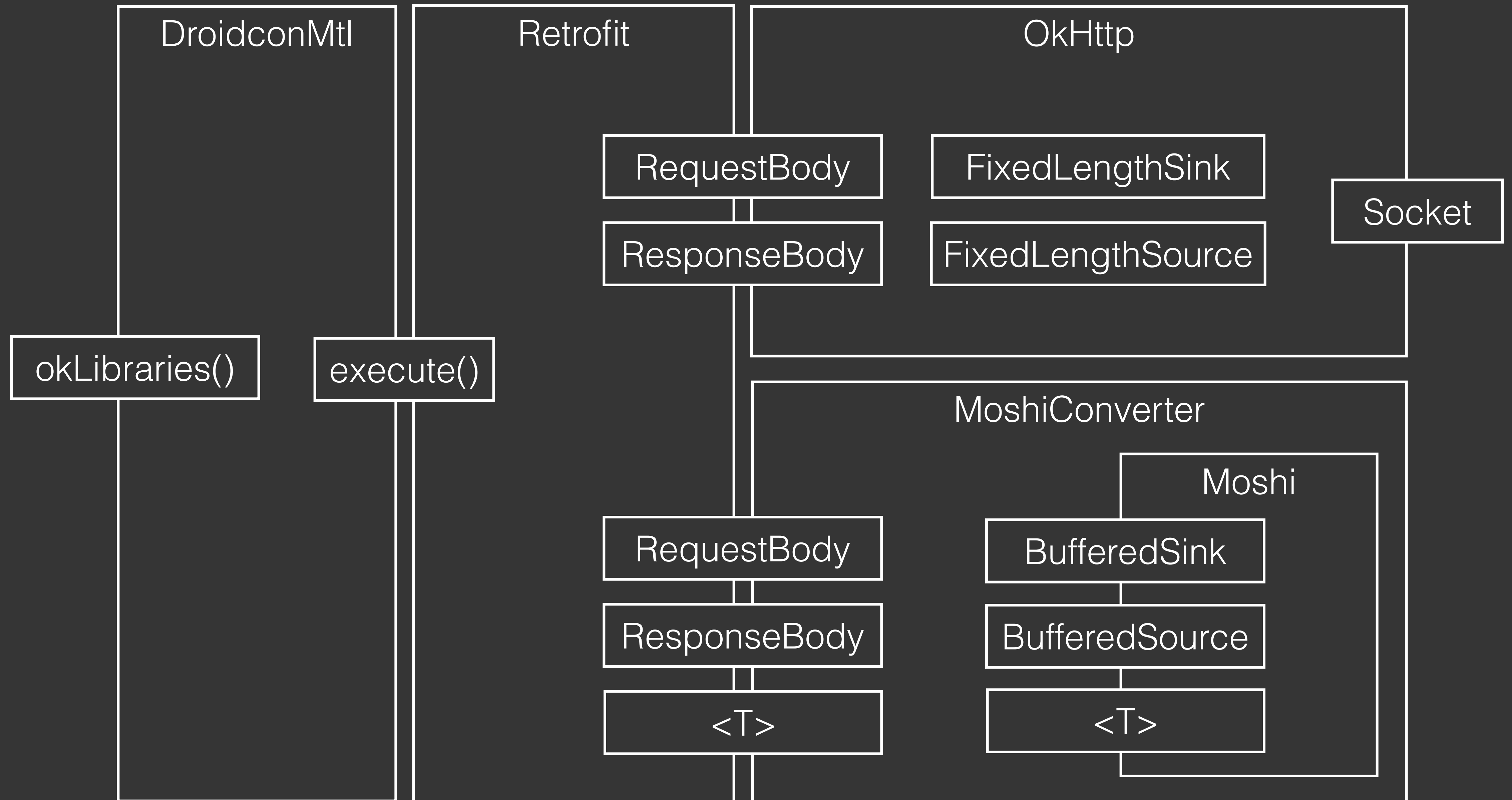


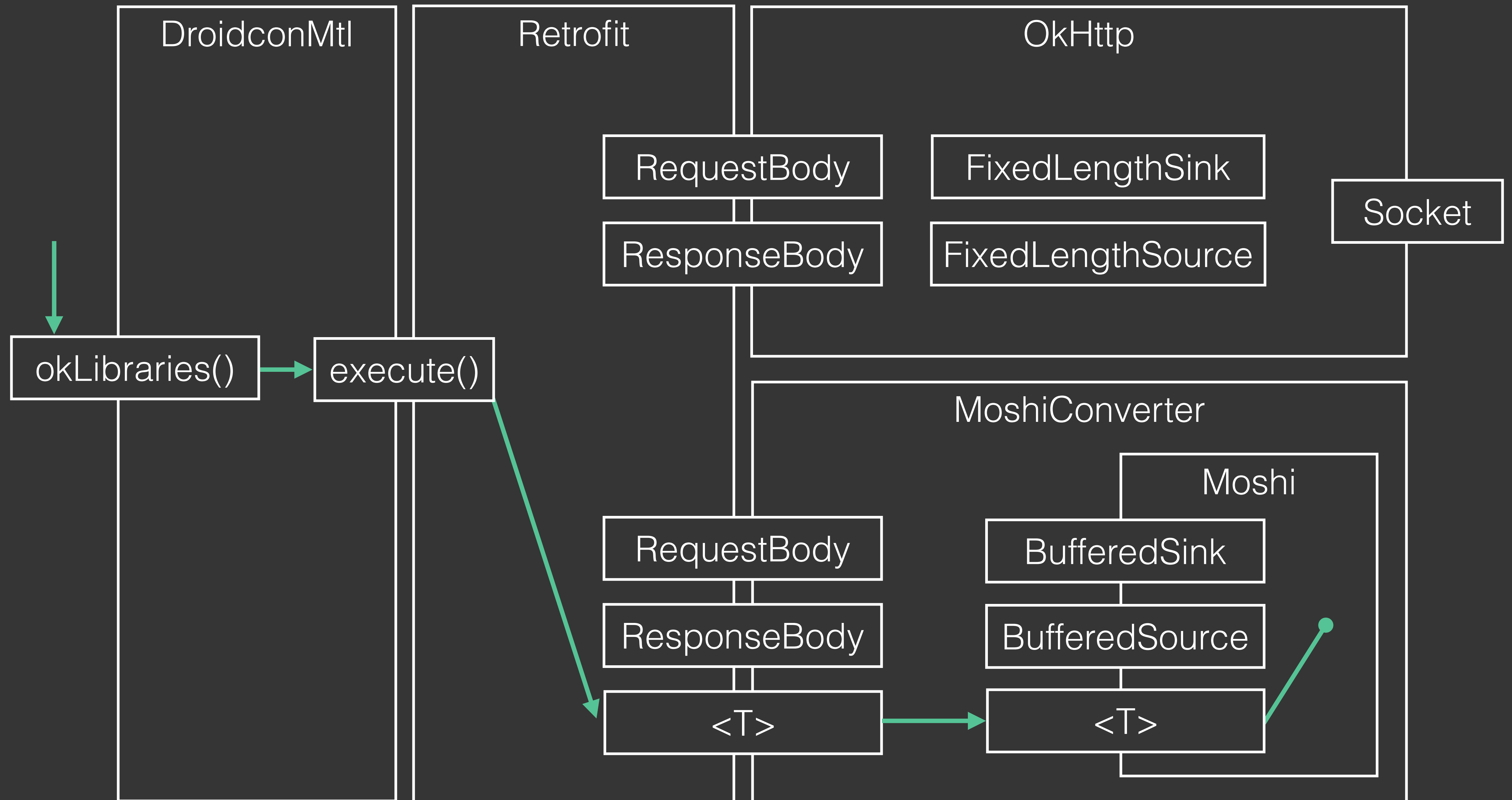


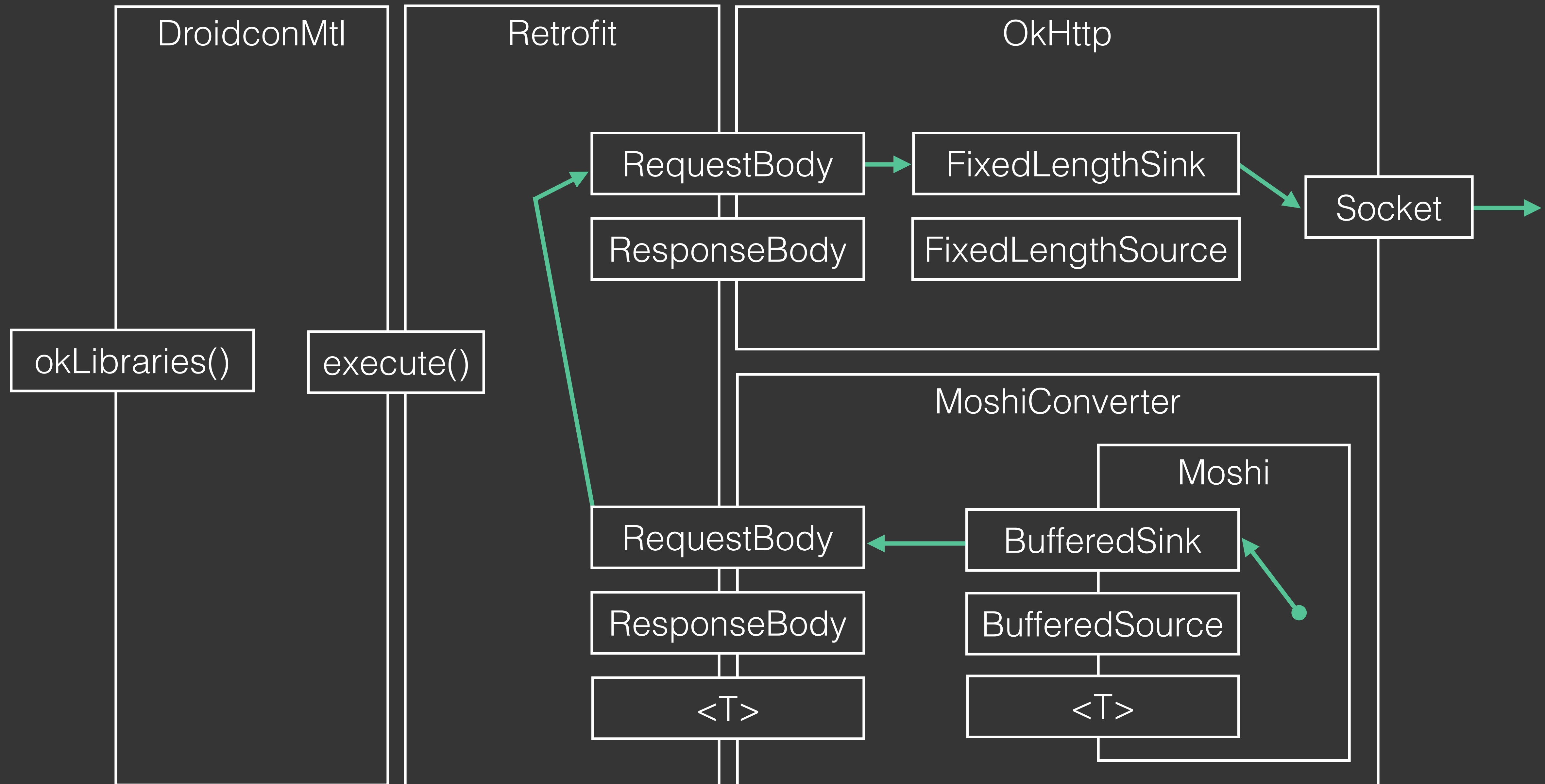


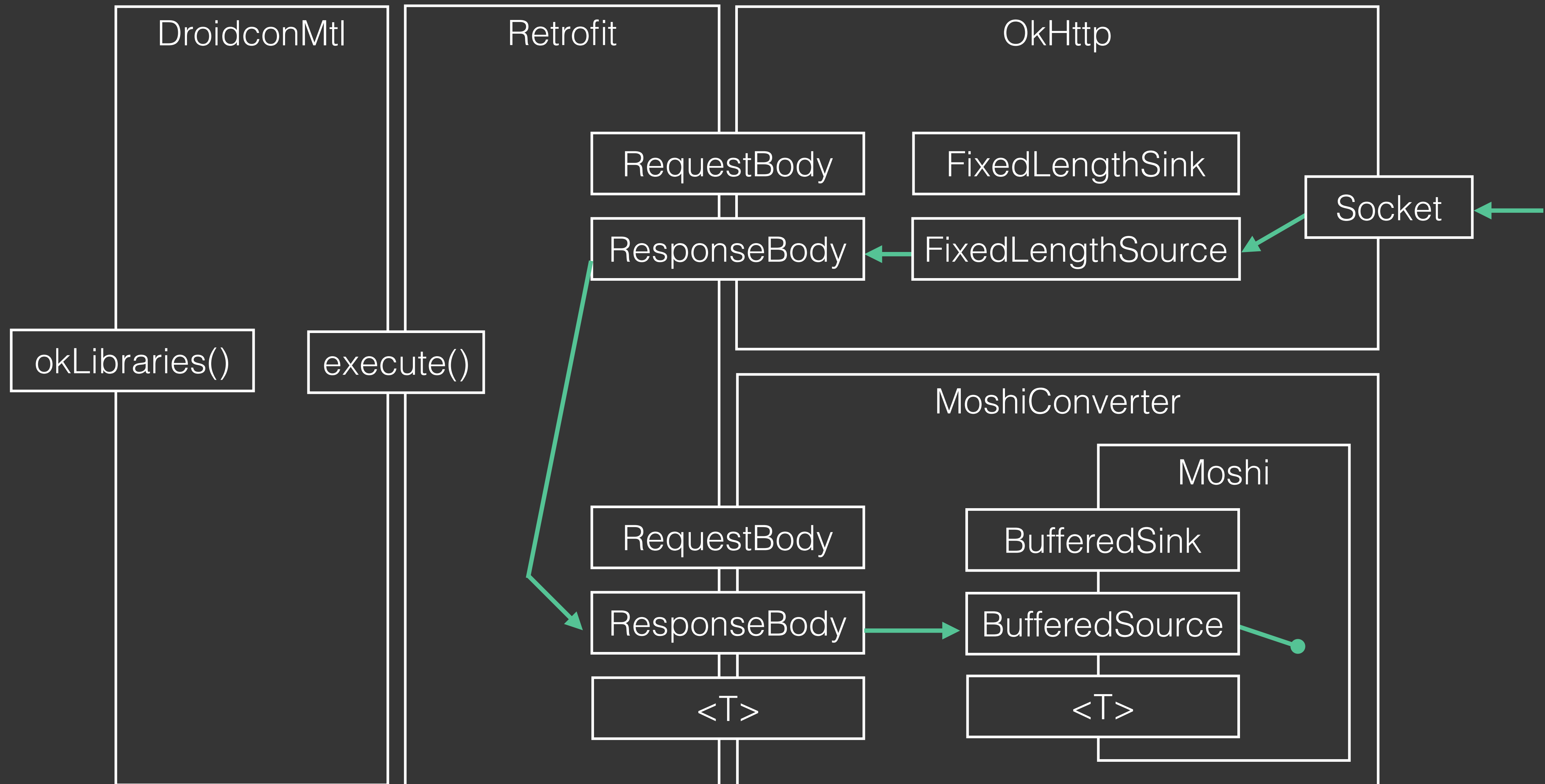


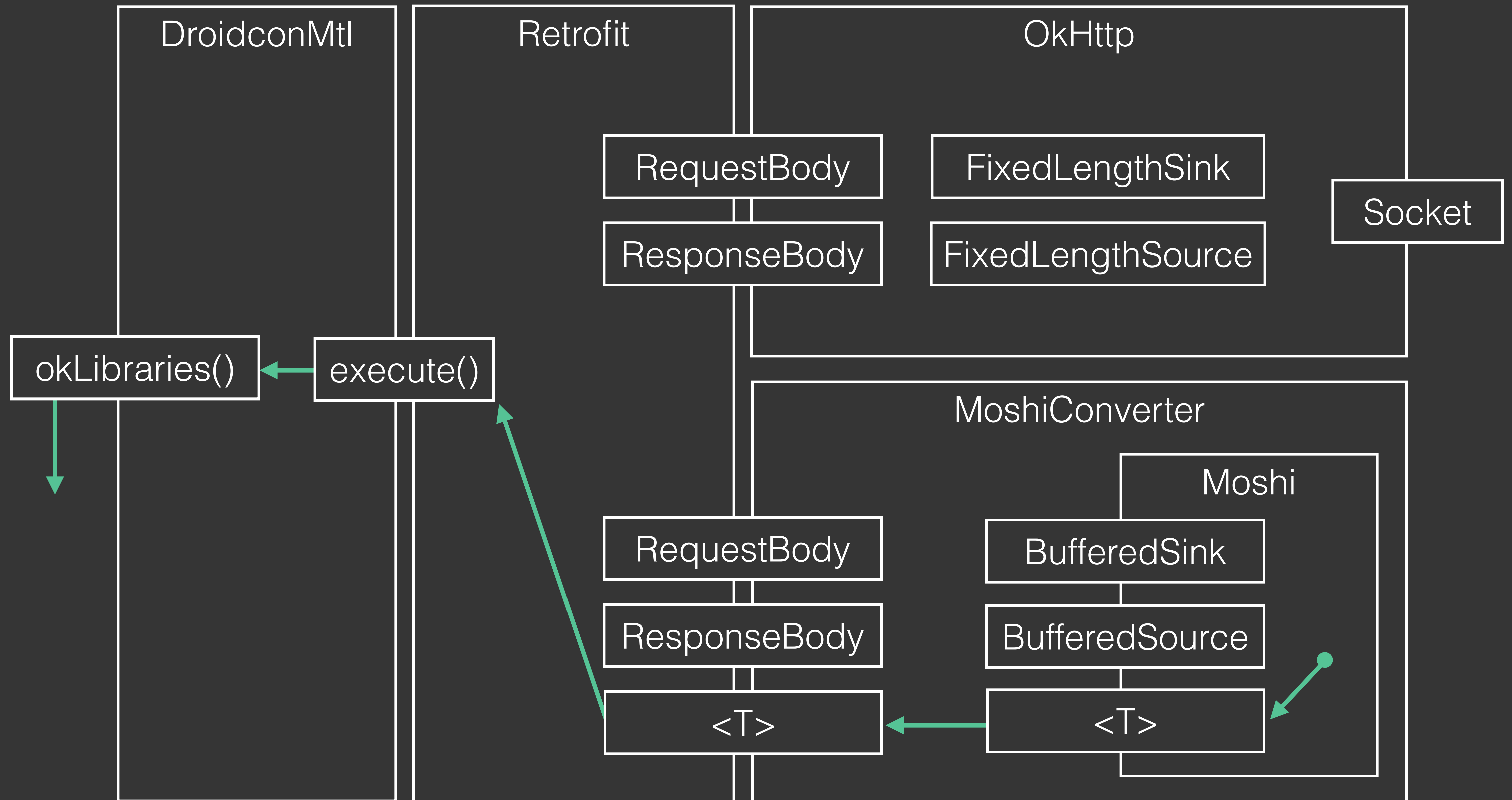


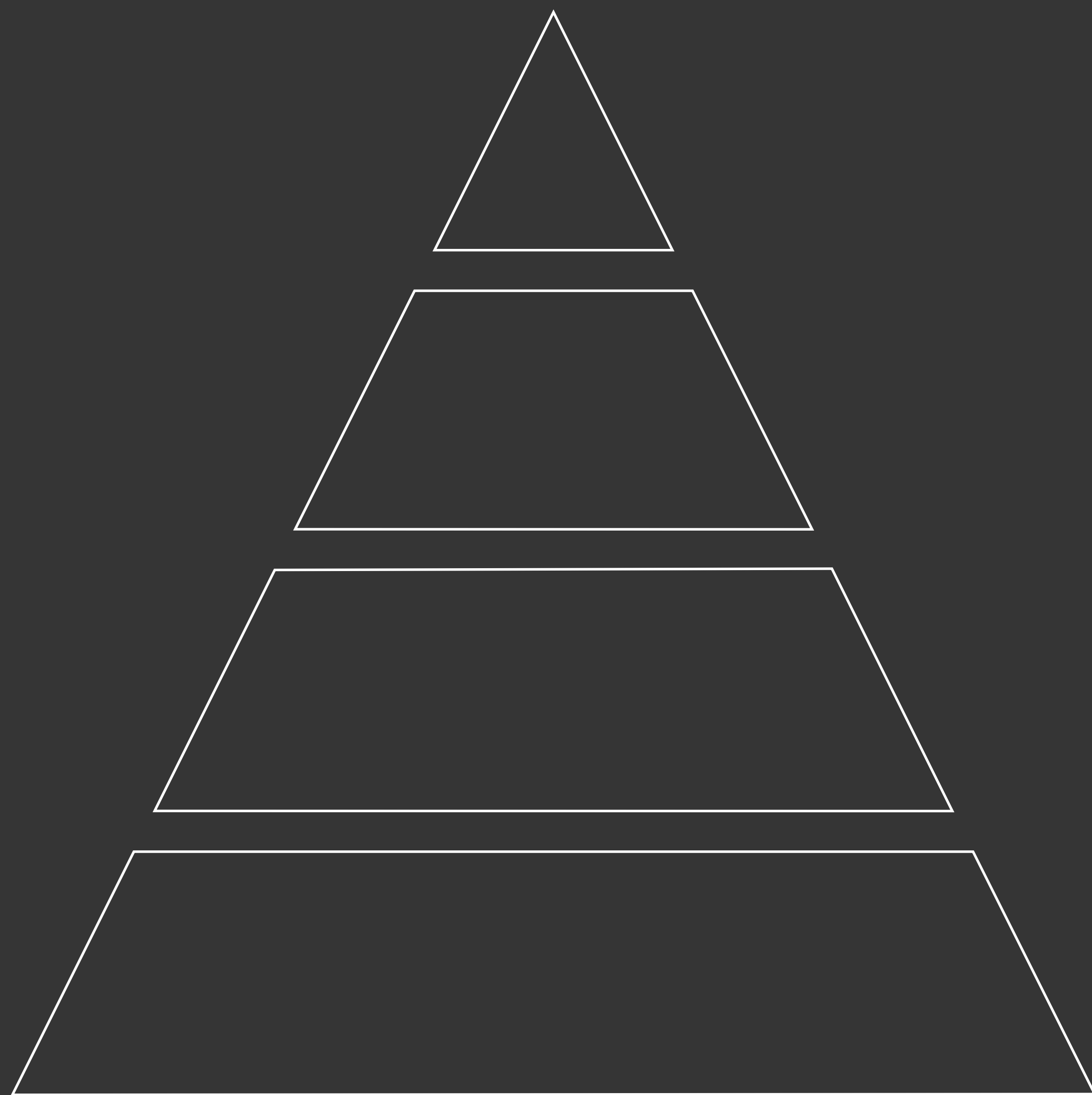












retrofit

moshi

okhttp

okio

A Few “OK” Libraries

A Few “OK” Libraries

[twitter.com/ jakewharton](https://twitter.com/jakewharton)

[google.com/+ jakewharton](https://google.com/+jakewharton)

jakewharton.com