dm-stripe
=========

Device-Mapper's "striped" target is used to create a striped (i.e. RAID-0) device across one or more underlying devices. Data is written in "chunks", with consecutive chunks rotating among the underlying devices. This can potentially provide improved I/O throughput by utilizing several physical devices in parallel.

```
Parameters: <num devs> <chunk size> [<dev path> <offset>]+
    <num devs>: Number of underlying devices.
    <chunk size>: Size of each chunk of data. Must be a power-of-2 and at
                  least as large as the system's PAGE_SIZE.
    <dev path>: Full pathname to the underlying block-device, or a
                "major:minor" device-number.
    <offset>: Starting sector within the device.
```

One or more underlying devices can be specified. The striped device size must be a multiple of the chunk size and a multiple of the number of underlying devices.

Example scripts
===============

```
[[
#!/usr/bin/perl -w
# Create a striped device across any number of underlying devices. The device
# will be called "stripe_dev" and have a chunk-size of 128k.

my $chunk_size = 128 * 2;
my $dev_name = "stripe_dev";
my $num_devs = @ARGV;
my @devs = @ARGV;
my ($min_dev_size, $stripe_dev_size, $i);

if (!$num_devs) {
        die("Specify at least one device\n");
}

$min_dev_size = `blockdev --getsize $devs[0]`;
for ($i = 1; $i < $num_devs; $i++) {
        my $this_size = `blockdev --getsize $devs[$i]`;
        $min_dev_size = ($min_dev_size < $this_size) ?
                        $min_dev_size : $this_size;
}

$stripe_dev_size = $min_dev_size * $num_devs;
$stripe_dev_size -= $stripe_dev_size % ($chunk_size * $num_devs);

$table = "0 $stripe_dev_size striped $num_devs $chunk_size";
for ($i = 0; $i < $num_devs; $i++) {
        $table .= " $devs[$i] 0";
}

`echo $table | dmsetup create $dev_name`;
```

]]